

Research Article

Multi-CrackNet: A Fast Segmentation and Quantification Combined Method for Rock Discontinuity Traces by Capturing an Image

Lan Qiao,^{1,2} Nai-Fu Deng ,^{1,2} Qing-Wen Li ,^{1,2} Qing-Long Zhang ,^{1,2}
and Jia-Wang Hao ^{1,2}

¹Department of Civil Engineering, University of Science and Technology Beijing, Beijing 100083, China

²Beijing Key Laboratory of Urban Underground Space Engineering, University of Science and Technology Beijing, Beijing 100083, China

Correspondence should be addressed to Qing-Wen Li; qingwenli@ustb.edu.cn

Received 1 May 2022; Revised 24 October 2022; Accepted 3 November 2022; Published 12 November 2022

Academic Editor: Yuanyuan Zha

Copyright © 2022 Lan Qiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Discontinuity investigation and characterization onsite is a labor-dependent work because current techniques cannot precisely handle multiple discontinuity identifications automatically under different work conditions. This paper proposes the multi-CrackNet which enables us to identify and segment linear discontinuities (joints and cracks) for random types of rock surface. A modified feature extraction network called the multiscale feature fusion pyramid network (MFFPN) has been developed based on FPN to capture and fuse more sensitive texture features of cracks across different types of background. With the help of a new training scheme by setting up 3 stages of training to simulate the human-based learning process, the established model can learn more features steadily and robustly from well-labelled databases. Additionally, a hybrid pixel-level quantification method is proposed to automatically compute the length, width, and inclination of cracks. Results show that the proposed method can achieve a detection accuracy of 87.1% for 1 to 9 sets of cracks on the rock surface across different types of rock. Case studies in Anshan West are provided to verify the reliability and accuracy of our method in macrolinear discontinuity identification and quantification, which sees great potentials in site investigation by saving a large amount of labor force.

1. Introduction

According to the “Suggested Methods for the Quantitative Description of Discontinuities in Rock Masses” proposed by the Standardization Committee for Laboratory and Field Tests of International Society for Rock Mechanics (ISRM) in 1978, the discontinuity of rock mass is mainly composed of joints and faults. Discontinuities indicate the boundary between interlayers in the rock mass and reveal the weak zones and the directions of fracturing of the rock mass under loading. The strength of rock mass in macroscopic scale is mainly determined by the strength of discontinuities and their combination forms. Therefore, in actual rock engineering, the primary task is to carry out sophisticated site investigation and research on the characteristics of rock discontinuous and to

clarify the engineering geological characteristics and spatial distribution law of rock discontinuous. Specifically, the fine-grained characterization of geometric parameters of complex rock discontinuities is vital to the geological investigation of engineering rock masses. Traditional visual inspection of rock discontinuities is generally operated by observing and manual recording with geological compasses and measuring tapes, which is time-consuming and labor-intensive [1–3]. Additionally, restricted by terrain, traditional survey methods with compasses or tapes are only applicable to accessible places. Meanwhile, traditional geological statistics is inevitably accompanied by margin of errors due to environmental or human bias.

Therefore, it is vital to bring down manual demands and seek better ways on discontinuity survey to improve the

investigation level by developing noncontact inspection methods. Over the past decades, vision-based methods with the development of photogrammetry and digital image technology for the discontinuity inspection of fractured rocks have been greatly developed and utilized [4–7]. Many remote sensing techniques such as digital photogrammetry, terrestrial laser scanning system, and unmanned aerial vehicles with high-resolution cameras can acquire 3D point clouds information so as to interpret the discontinuity planes or linear discontinuities [8–11]. With the help of accurate 3D point cloud information, it is indeed feasible to acquire geometrical features of discontinuities such as orientation, spacing, persistence, number of sets, and block size mentioned in ISRM in 1978 [12, 13]. To reduce manual workload in site investigation, semiautomatic or automatic program based on 3D point cloud techniques is developed to conduct regional discontinuity analysis. For instance, Li et al. [14] proposed an automatic method for trace mapping based on normal tensor voting theory. Guo et al. [15] also developed a new approach based on 3D point cloud to extract trace information from outcrops. Kong et al. [16] provided a hybrid method to identify and extract four discontinuity parameters from 3D point cloud, namely, number of sets, orientation, spacing, and trace length. However, these techniques heavily rely on the sophisticated equipment to capture precise spatial information of the outcrop terrain, allowing comprehensive interpretation from different aspects.

The detection of discontinuities on the rock surface is extremely challenging via simple photography-based technology, especially at the pixel-wise level, because the complex geometric shapes and morphologies hinder the performance of typical object detection methods. Current research mostly considers all linear discontinuities as proper joints without any displacement between interlayers, which ignores the width of linear discontinuities. This paper aims at identifying linear discontinuities (joints or cracks) by considering their real size and real shape; this means our detection task requires pixel-level performance to achieve in situ picking-up of discontinuity features. To this point, deep convolutional neural networks (DCNNs) have been adopted to deal with such detection task on rock surfaces. Current research adopting DCNNs in civil engineering mostly emphasizes the detection and maintenance of civil infrastructure, including concrete crack detection [17–20], rust inspection of iron structure [21], loosen state of bolts [22], sewerage leakage determination [23, 24], and pavement health detection [25, 26]. These applications have greatly improved the working efficiency of defect detection and made significant contributions to engineering informatization. However, these studies rarely aim at the image acquisition scenarios of rock discontinuities.

In this paper, we aim to carry out fine-grained identification and quantification of rock linear discontinuities (cracks) on various types of rock surfaces and segment each crack from the background at pixel level. Current batch-based segmentation methods have provided excellent pixel-wise segmentation methods for rock defect detection using deep learning, but they mostly consider simple weak interlayer

segmentation or tunnel face defect identification [27–29]. These methods can barely handle the fine-grained segmentation problem of multiple linear discontinuities at the pixel level. Therefore, this paper provides an approach to process rock images captured by normal cameras or cell phones and provides the multi-CrackNet framework based on mask R-CNN to identify each crack from multiple intersecting interlayers. The method mainly contributes to the following three points:

- (1) Deep learning-based method requires larger training dataset; thus, this paper proposes an image augmentation method to enlarge the dataset from 257 images to 4369 images, and these images are then labelled manually at pixel-level granularity to feed to the multi-CrackNet. Furthermore, a sequential training method is proposed based on human-based learning process to increase the accuracy of segmenting each staggered discontinuity from each other
- (2) Multi-CrackNet, developed based on mask R-CNN [30], proposed a special feature pyramid network denoted as multiscale feature fusion pyramid network (MFFPN) which merges a hierarchy of different features obtained by different convolutional cores and provides better combination feature maps to the next stage
- (3) An intelligent quantification method is proposed based on the distance transfer method (DTM) to efficiently compute the length and maximum width of each segmented linear discontinuity. Our method combines a variety of image process techniques such as binarization, skeletonization, corrosion, expansion, and DTM. This fused quantification method can accurately characterize basic geometric information of the segmented cracks with robust performance
- (4) Case studies in Anshan West elucidate that the proposed method succeeds in denoting the number of cracks in seconds and segmenting 1 to 9 sets of staggered cracks with 87.1% accuracy on random types of rock surface across different environments. Additionally, this hybrid method shows great performance of quantification of linear discontinuities on a selected outcrop, which sees potential onsite investigation with handy portable equipment such as smartphones

2. Related Work

The past few years have seen great progress in research of vision-based crack detection. Typically, existing methods for rock crack identification can be classified into the following categories: (1) the first category is based on grayscale intensity threshold, which assumes that the crack and the noncrack background have obvious grayscale differences. Many reported methods for threshold optimization involve

global thresholds [31], local thresholds [32], and the self-adaptive threshold method using the Otsu algorithm [33]. These methods are relatively fast in processing simple images with less noise but yield poor performance for complicated scenarios and even fail in crack segmentation. (2) The second category is based on edge detection methods such as Gabor filters [34], Laplace filters [35], and Gaussian filters [36], which pay attention to the boundary between the background and the cracks. These methods perform better than grayscale-based methods but can barely handle a single crack segmentation. (3) The third category relies on machine learning and deep learning techniques. The convolutional neural network (CNN) is well known for feature extraction and producing promising results in crack detection [37]. CNN-based methods normally detect the objects from patches of pixels obtained from the feature map. These methods can only determine whether or not the patch contains the object (crack) or the presence of the object in the image. However, a comprehensive analysis shows that many engineering works require not only the object detection but also accurate semantic segmentation of the objects; for example, the detection problem requires pixel-level segmentation of each crack from the background, which can further provide robust data support for in-depth geological analysis.

Pixel-level crack segmentation is substantially affected by the number of cracks and the layout of cracks. Many efforts have been made over the past few years to deal with single crack segmentation or interlayer segmentation, but very limited research has focused on multicrack segmentation, namely, identifying and segmenting each crack among various sets of discontinuities from the background. It is very challenging for AI training because most AI models such as fully connected network (FCN) [38], faster R-CNN [39], and mask R-CNN [40] are good at extracting features from a particular shape of a type of object. However, the shape of cracks is arbitrary and hard to describe by a set of particular features, which yields poor performances of crack segmentation because of the complicated geoenvironment and working conditions. Furthermore, crack detection is heavily affected by noise and lights; existing methods rely on preprocessed approaches to distinguish known features between cracks and background noises, such as numerical features in segmented patches [29] and multiple greyscale filters [41]. However, the above methods require prior knowledge of different types of cracks, which means that the obtained features can be barely generalized to various types of cracks. Some postprocessed techniques such as Bayesian decision theory are utilized to filter out falsely detected cracks by eliminating the impact of noises from similar types of backgrounds, environments, or working conditions [34]. These methods use very limited pixel-level information obtained from the images to identify noise or noncrack pixels, which can hardly handle complex crack detection conditions across different types of environments. In this paper, the proposed multi-CrackNet with MFFPN structure to further refine the segmentation results from different complex backgrounds by aggregating different scales of pixel-level features accompanied with a special training scheme, which enhances the learning ability of the detection model.

3. Geological Background

Anshan West is located at southwest of Anshan city in Liaoning province of China. It is an abandoned open-pit region with complex geological structure as indicated in Figure 1. This site is selected as the region of interests due to the recovery need for habitation. Because its distinctive geological conditions and the columnar structures are prone to rockfall or toppling, it is necessary to carry out detailed site investigation before the launch of any engineering project. The joint is a structural plane formed by rock mass under the action of stress. It is a kind of tectonic fracture with no displacement or very small displacement. Although the extension of joints is not frequent and the depth development is not large, they dominate the majority. Thus, the stability, failure mode, and failure process of the ore body and its surrounding rock mass are controlled by the orientation, quantity, size, and morphology of the joints. At the same time, as a kind of tectonic movement, joints can also reflect the tectonic outline of the region, which provides basic data for the mechanical analysis of regional tectonic stress field and tectonic system.

To master the quality and stability of rock mass in our study area, we carried out a detailed investigation on surface joints and collected the strike, the dip, the inclination, the spacing, and filling conditions of joints. The general distribution information of joints is listed in Table 1. There were 1471 groups of joint statistical data with an average spacing of 32.3 cm. According to the dip analysis, there were 4 groups of dominant orientation, accounting for 70.17% of the total number of joints. The first group had a dip of $10^{\circ}\sim 50^{\circ}$, accounting for 14.67% of the total number of joints, with an average dip of 29.57° and an average dip angle of $64.59.05^{\circ}$. The second group had a dip of $90^{\circ}\sim 160^{\circ}$, accounting for 30.53% of the total number of joints, with an average dip of 124.00° and an average dip angle of 61.11° . The third group had a dip of $190^{\circ}\sim 230^{\circ}$, accounting for 12.87% of the total number of joints, with an average dip of 210.72° and an average dip angle of 65.96° . The fourth group had a dip of $290^{\circ}\sim 330^{\circ}$, accounting for 12.10% of the total number of joints, with an average dip of 310.67° and an average dip angle of 54.82° . According to the strike analysis, there are two groups of dominant orientation which are 35° and 295° , and the inclination is mainly distributed in $57^{\circ}\sim 66^{\circ}$. The distribution characteristics of joints are shown in Figure 2.

During investigation, a fundamental issue for characteristic analysis of rock mass is to efficiently acquire the trace length and width of the joints. Manual acquisition of these parameters relies on tedious and repetitive notation work (see Figure 3). Although current cloud-point-based techniques can perform well on characterizing joint features, they mostly rely on high-end instruments or special onsite setting to know the spatial information of the joints. In this paper, we develop a hybrid method to accurately identify and segment each joint out of complex fractured rock surface from ordinary images taken by a smartphone or any other handy equipment. Our method can automatically compute the trace length and width in pixel level regardless of scale or types of rock. Detailed methodology will be elaborated in the following chapters.

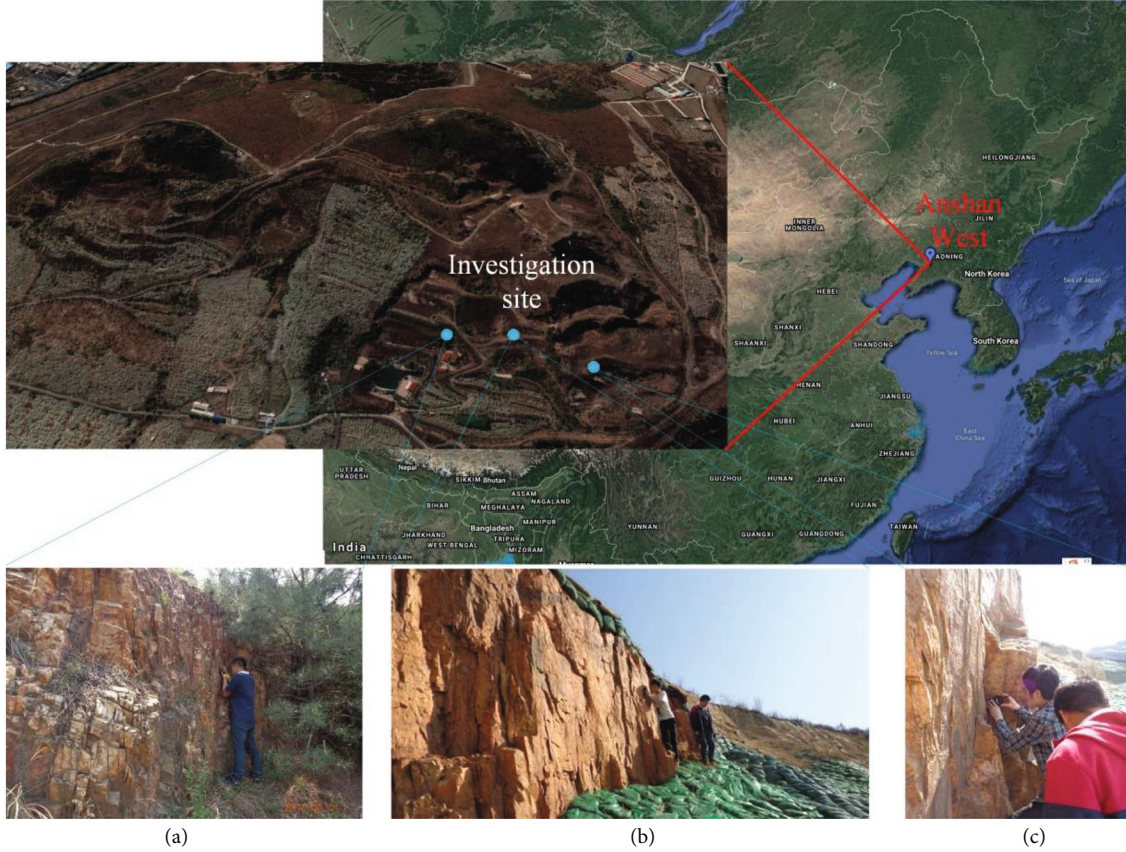


FIGURE 1: Geographical location of the study area and site work. (a), (b), and (c) are manual investigation with geological compass and tape measure.

TABLE 1: Distribution of joints in the study area.

Outcrop length (m)	Number of joints	Average spacing (cm)	Number of joints containing filling	
			Rigid	Soft
475.8	1471	32.3	120	60

4. Methodology

In this paper, we have divided our task into 4 parts, namely, the image acquisition and annotation, crack identification and segmentation, crack classification, and onsite validation. Initially, an image augmentation method is introduced first to enlarge our initial database to meet the requirements of deep-learning-based methods. Accompanied with a new proposed training scheme based on the progressive learning process, the constructed deep learning framework can produce desirable performance. The proposed deep learning framework is called the multi-CrackNet which is based on mask R-CNN with several improvements, especially targeting at rock crack (joint) detection under multiple working conditions, detailed elaboration is explained in Section 4.2. To quantify and characterize rock cracks after segmentation, a well-design size measurement method is proposed based on the distance transfer method (DTM) to automatically calculate the length, width, and inclination of each crack. Finally, case studies from Anshan West are performed to prove

the validation of our method in site investigation. As shown in Figure 4, multi-CrackNet is developed based on mask R-CNN to localize and segment each crack from the background at pixel level. The segmented cracks are then fed into the quantification framework with hybrid methods to derive the length and width of the crack into pixels.

4.1. Data Collection and Annotation. Images of rocks with fissures and joints vary greatly due to many factors such as rock types, source regions, capture means, and sampling window size; thus, crack recognition for random scenes is barely possible. Furthermore, there exist multiple interlaced cracks within a single sampling window. Normally, inter-layered cracks create difficulties in segmenting each crack; current research focuses on the segmentation of all cracks within a single sampling window from the background, which indicates that the fine-grained segmentation of a single crack from a set of interlaced cracks is challenging. To deal with the above problem, the goal of this paper is to train the model with various types of rock on different scales and

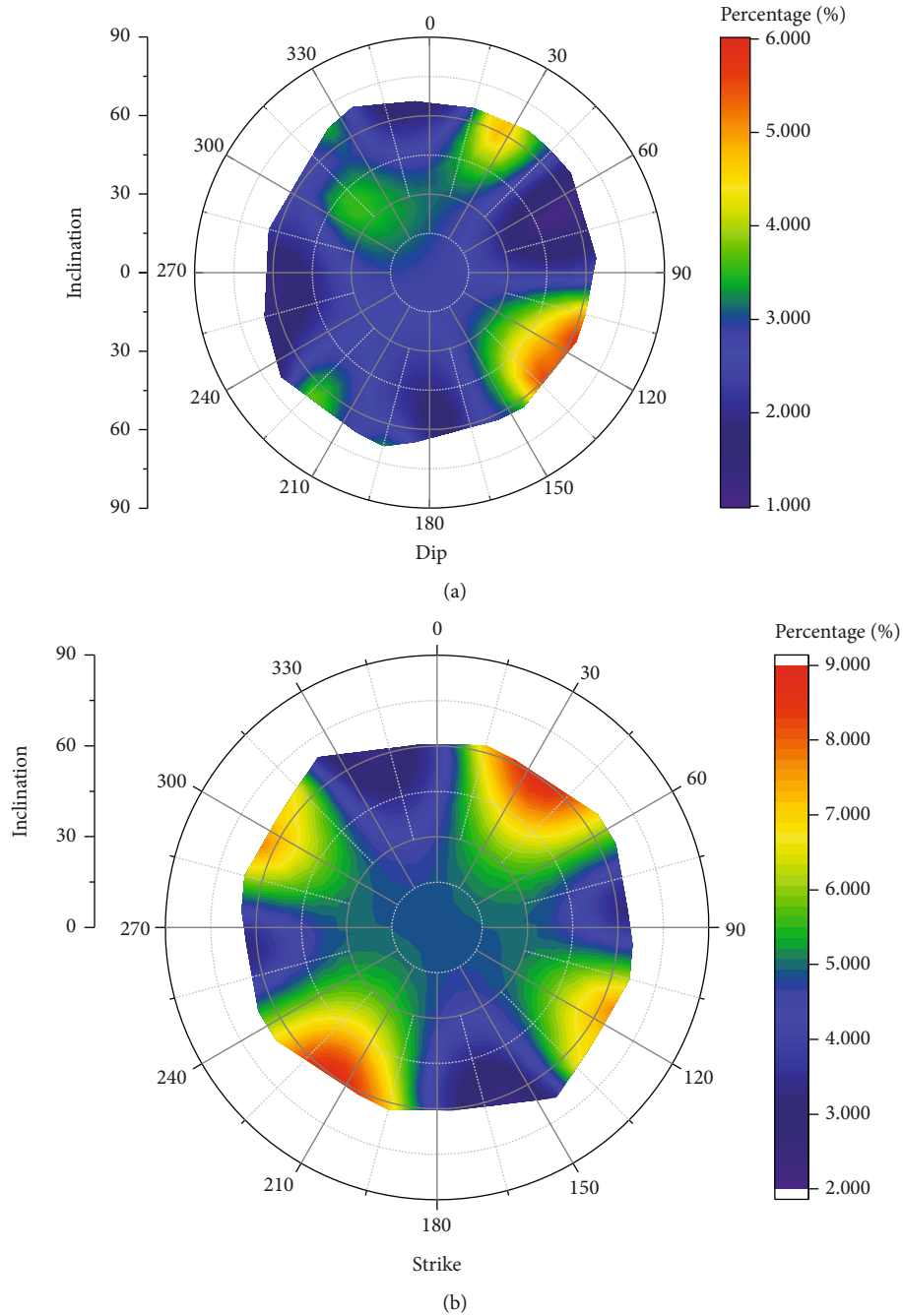


FIGURE 2: Pole density plot of dip and strike with inclination. (a) The dip angle of discontinuities; (b) the strike angle of discontinuities.

achieve fine-grained crack segmentation for different types of working conditions. Therefore, the database appears to be particularly important.

Initially, 257 images with size 1024×1024 were captured from different types of regions to build the primary training database S0. These raw images are then cropped into 16 small 256-by-256 images which form the secondary database S1 with $257 \times 16 = 4112$ small images. According to the number of cracks in each small image, the paper manually classified the secondary database into three subdatabases, that is, S1-1, S1-2, and S1-3. S1-1 contains images with only 1 set of cracks;

S1-2 possesses images with 2-3 sets of cracks; S1-3 covers images with more than 3 sets of cracks. Then, this paper uses LabelMe [42] to annotate each crack on each image to finalize the formation of dataset S1-1, S1-2, and S1-3.

Figure 5 elucidates the transfer-learning process of training the multi-CrackNet; specifically, a pretrained model from the COCO dataset [43] is used to train subdatabase S1-1; the output model weight W1 is adopted to train subdatabase S1-2 subsequently; after that, the output model weight W2 is applied to trained the subdatabase S1-3 with model weight W3 as the final output.

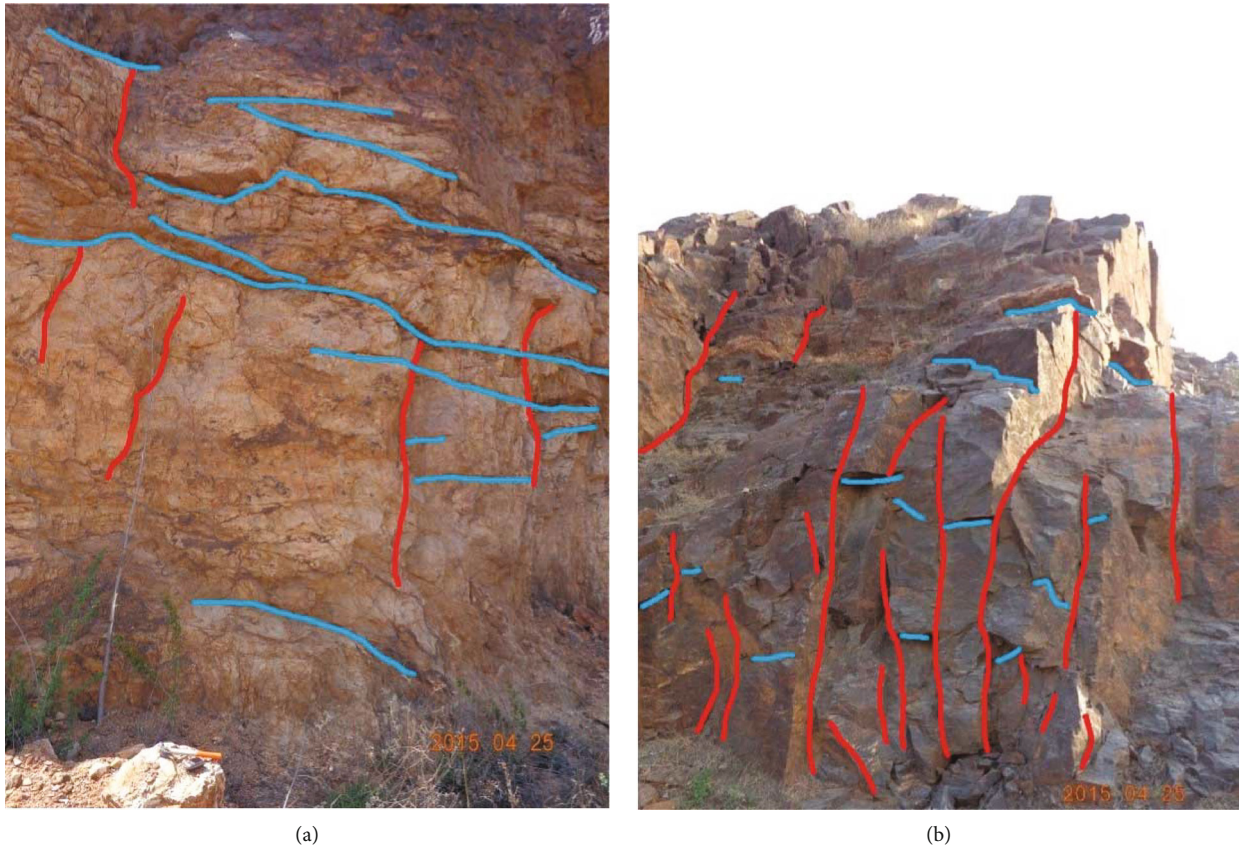


FIGURE 3: Tedious and tiring manual detection of discontinuities in site investigation. (a) and (b) are examples of manual labelling of discontinuity traces.

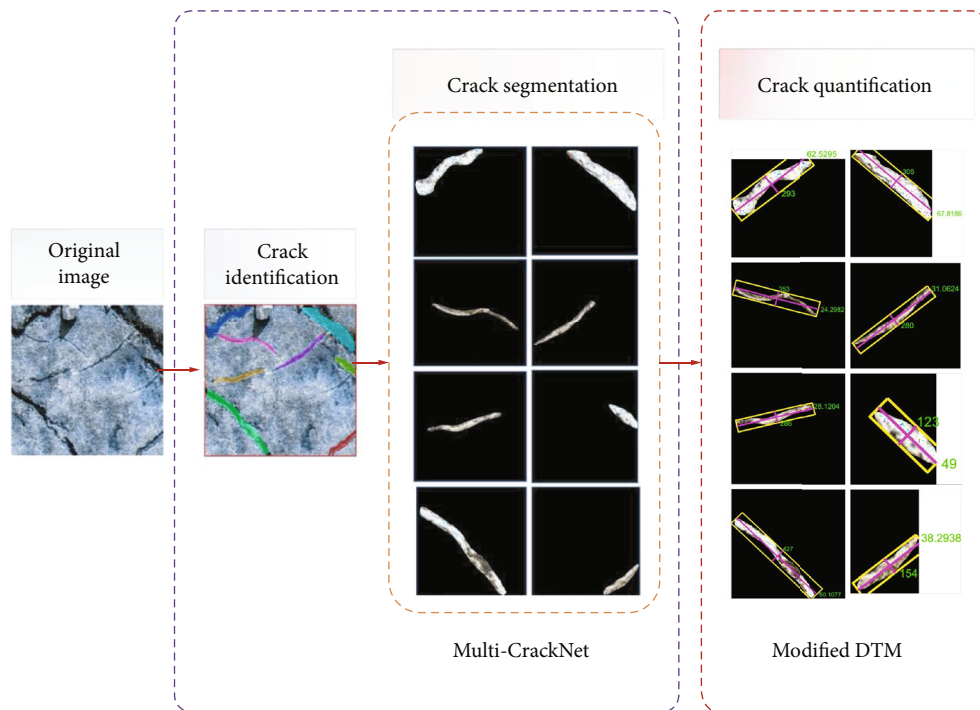


FIGURE 4: Overview of the hybrid multicrack identification and quantification method.

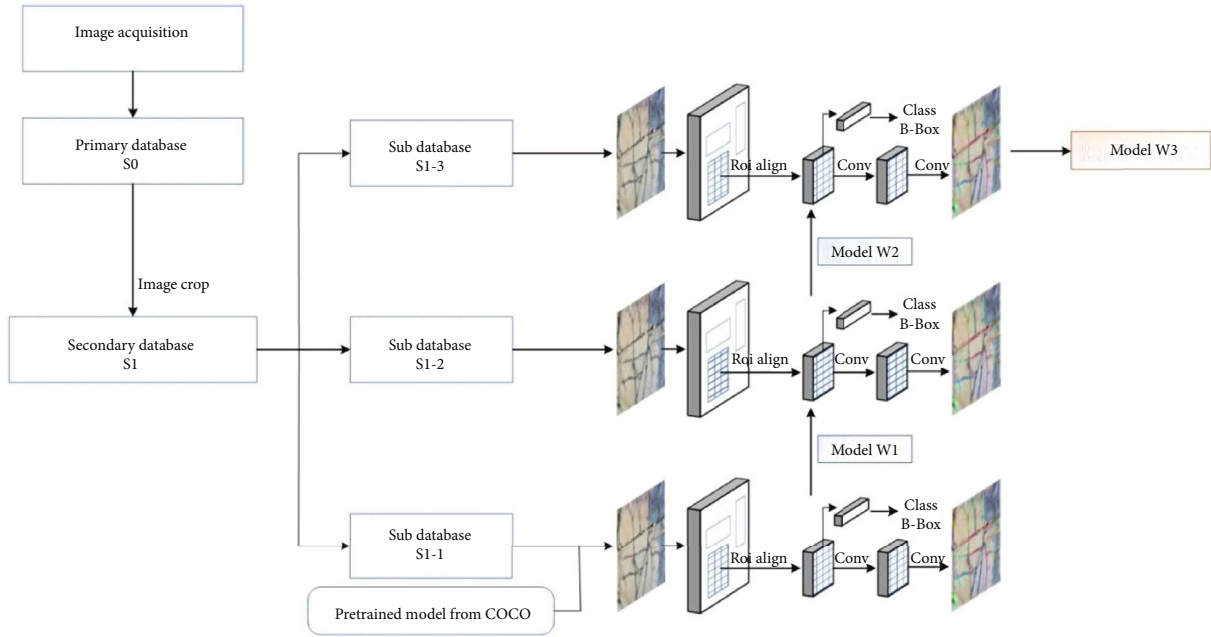


FIGURE 5: Operation schematic diagram of crack detection (the training of model weight W1 to W3 is contributed by the same framework of multi-CrackNet).

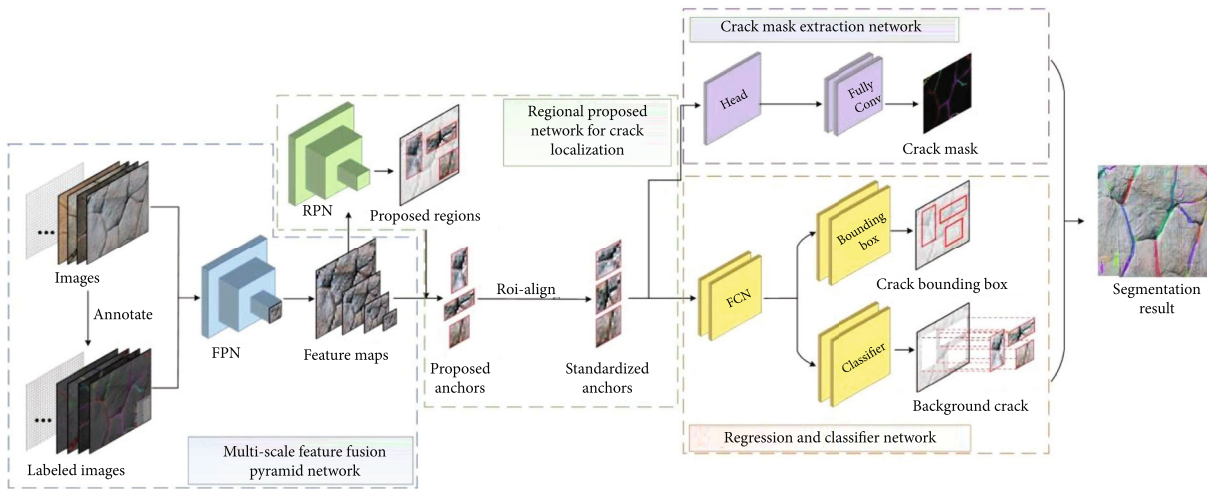


FIGURE 6: Detection framework of multi-CrackNet.

4.2. Crack Detection Framework. The modified crack detection framework is multi-CrackNet based on mask R-CNN which utilizes the advantages of many state-of-the-art algorithms through combination. The framework of multi-CrackNet is presented in Figure 6. Similar to mask R-CNN, the framework architecture involves 4 parts, namely, a multiscale feature fusion pyramid network (MFFPN) backbone, a regional proposal network (RPN), the classifier and bounding box regression branches from faster R-CNN, and a fully convolutional network for instance segmentation.

4.2.1. Multiscale Feature Extraction and Fusion of Cracks. The mask R-CNN algorithm was developed for multiclass object detection across different application environments; in this paper, to deal with rock crack detection in complex

scenarios, an improved feature extraction network is designed based on the traditional feature pyramid network (FPN) to capture and merge more levels of texture features of rock cracks. The uneven scale of natural defects on rock surfaces brings great challenges to the identification of cracks due to the scale of effect. To solve this problem, the multiscale feature fusion pyramid network (MFFPN) aims at stratifying and extracting different scales of crack features from images based on an FPN framework presented in Figure 7. Convolutional neural network (CNN) can extract different features from the target image matrix by using different convolutional kernels. For machine vision, image features are mainly embodied as texture features, namely, flat textures, vertical textures, and inclined textures of joint and fissure correspondingly. To project the texture features of

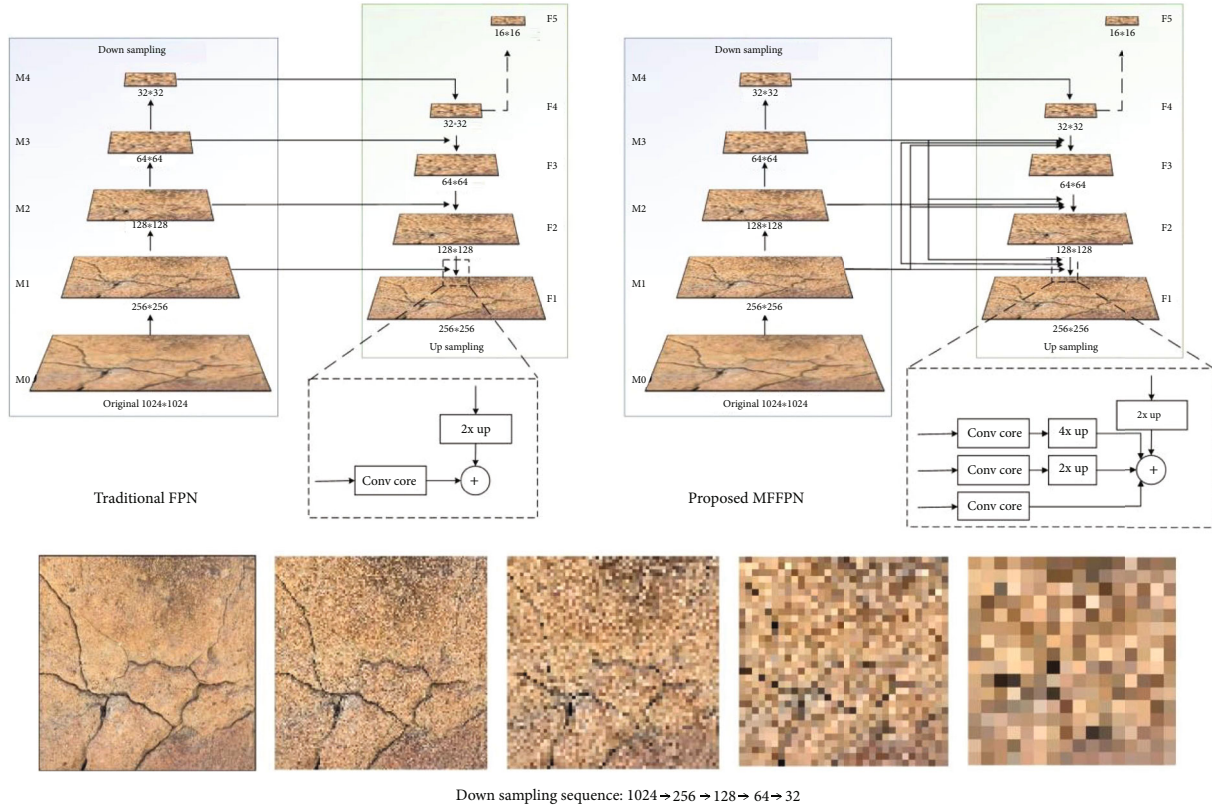


FIGURE 7: Comparison between traditional FPN and the proposed MFFPN.

different cracks onto the feature mapping layer, this paper proposed a fusion network that combines multiple convolutional cores in an FPN architecture. Specifically, the MFFPN is based on an FPN architecture with 4-layer downsampling (M1~M4), 4-layer upsampling (P1~P4), and a subsampling layer P5 from P4; each downsampling layer contains multiple convolutional operations with different convolutional cores. The convolutional results are then concatenated together to feed on the upsampling structure; the upsampling layers will fuse all convolutional results from each convolutional core and then proceed with the upsampling operation, passing all texture features through the whole upsampling stream to maintain as much texture information as possible during the upsampling operation. For instance, in Figure 7, the traditional FPN in layer F1 only concatenates the convolutional results from layer M1 and the upsampling results from layer F2, which neglects crack features in other scales. Our modified MFFPN utilizes all convolutional results from layers M1, M2, and M3 and then concatenate them with the upsampling result from layer F2. In this manner, our feature extraction network can maximize the retention of crack characteristics at different scales.

In this research, rock defects are normally strip-shaped with crossing nodes; therefore, this paper selects 5 primary convolutional kernels that can reflect the characteristics of flat, vertical, inclined, and cross texture of the cracks; these primary convolutional kernels are then trained to adapt the shape of cracks. To be specific, for example, the M4 layer will go through multiple convolutional cores to extract different

types of texture feature; these features are then concatenated together to feed to F1 to F4; at the same time, other down-sampling layers from M1 to M3 will go through the same operation and seed concatenated features to F1 to F5. Each layer in the upsampling layer, such as F3, will receive 4 feature maps from M1 to M4, and these feature maps will be concatenated together with F3 to keep upsampling to F2. During this process, upsampling or downsampling operation (see Figure 7) will be applied to each feature map derived from M1 to M4 to ensure that the size is fitting to each upsampling layer (F1 to F4) so that concatenating operation can proceed. It is noticeable that this MFFPN framework turns out to be a prune-able network, which means that during the training or inference stage, we can apply pruning operation to cut off some convolutional cores that may lead to undesired results.

4.2.2. Location of Crack Positions. The MFFPN can obtain various texture information of multiscale cracks from the images, but the extracted information contains not only the texture information of cracks but also the information from the background. To narrow down the detection regions and localize the position of cracks, the RPN was adopted to generate proposed bounding boxes for each image to indicate the position of cracks. The RPN relies on a shared sliding windows on the convolutional feature maps (F1~F5) passing from MFFPN to generate 15 pre-designed anchors on each pixel; these anchors contain 5 kinds of size with 3 types of ratio. Different sizes of anchors will be applied to

different layers of the feature map from F1 to F5, respectively; to localize different scales of cracks, different ratios of anchors are utilized to adapt different shapes of cracks.

Specifically, the anchor sizes are designed as $(48^2, 96^2, 192^2, 384^2, \text{ and } 768^2)$ and the anchor scales are set to be $(1:4, 1:1, \text{ and } 4:1)$. As indicated in Figure 8, the sliding window will go through each pixel on the convolutional feature maps generated from MFFPN and create 15 anchors; these anchors are then used to compare with the ground truth bounding boxes of each crack by the Intersection over Union (IoU) algorithm to select the most likely anchors step by step. The IoU algorithm determines the similarity of the anchor box and the ground truth bounding box by simply calculating the overlapping area of the two boxes.

$$\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}}. \quad (1)$$

In this paper, we choose the IoU limit as 0.7, which indicates that the only anchors with an overlapping ratio greater than 0.7 can be retained; subsequently, those anchors whose IoUs of the same ground truth are greater than 0.7 should be compared via the nonmaximum suppression (NMS) method [44] to keep the most likely anchors corresponding to each crack. As seen in Figure 9, if the IoUs of more than two anchors over the same ground truth are greater than 0.7, we need to compare these anchors by calculating their IoU; if the IoU of these anchors is greater than 0.5, we need to abandon the one with the lower IoU over the ground truth; finally, the remaining anchors are output as proposals.

4.2.3. Classification and Segmentation of Cracks. In Figure 8, RPN utilizes proposals to localize the candidate positions of each crack; before entering the next stage, these proposed regions are fed to RoI align layers to resize to the proper size. RoI align is proposed by Gkioxari et al. and firstly applied in mask R-CNN [30]. Because the generation of proposals relies on convolutional operation in each layer from F1 to F5 and the size of each proposal is different, RoI align is introduced to resize and standardize the proposals to fit in the input requirement of fully connected network (FCN). After standardization, these fixed feature maps are concatenated together and fed to two branches of prediction which output the crack position and shape (see Figure 6). On one hand, the first branch of FCN flattens the feature maps to reveal and pass higher levels of semantic information to the regression and classification layers; on the other hand, the feature maps will go through another branch of the fully convolutional layer to predict the shape of crack.

Specifically, as depicted in Figure 10, the 7×7 feature maps will go through a fully convolutional layer to extract high-level semantic information of the proposals (RoI align), and the output convolutional feature maps will be flattened by two branches of FCN; the first FCN will pass the information to a softmax [45] operation to normalize the results as a class probability which reveals the confidence of crack detection with the proposals; those proposals with class probability higher than 0.75 are considered to be cracks (positive

anchors) while the others are considered to be the background (negative anchors).

The second FCN operation provides robust information to the bounding box regression, which trains the bounding box to approach the ground truth. It can be seen from Figure 9 that the proposed anchor box is slightly different from the ground truth, so a fine-tuning operation is needed [30]. If we suppose the coordinate of the ground truth is $G = [G_x, G_y, G_w, G_h]$, here, G_x and G_y are the coordinates of the center and G_w and G_h are the width and height of the ground truth box. Given that the position of the proposed bounding box is $\text{Bbox} = [B_x, B_y, B_w, B_h]$, then we are looking forward to a transformation that leads to

$$F(B_x, B_y, B_w, B_h) = (G'_x, G'_y, G'_w, G'_h), \quad (2)$$

where (G'_x, G'_y, G'_w, G'_h) should be as close as possible to (G_x, G_y, G_w, G_h) . In this paper, we assume that the transformation of bounding box is simply comprised of a translation and a scaling, which can be written as follows:

$$\text{translation} : \begin{cases} G'_x = B_w \cdot T_x(B) + B_x, \\ G'_y = B_h \cdot T_y(B) + B_y, \end{cases} \quad (3)$$

$$\text{scaling} : \begin{cases} G'_w = B_w \cdot e^{T_w(B)}, \\ G'_h = B_h \cdot e^{T_h(B)}, \end{cases} \quad (4)$$

where $T_x(B)$, $T_y(B)$, $T_w(B)$, and $T_h(B)$ are the transformation coefficients to be trained. Therefore, if we let $W = [T_x(B), T_y(B), e^{T_w(B)}, e^{T_h(B)}]^T$ and $\Phi(B) = [B_x, B_y, B_w, B_h]$, then

$$P(B) = W \times \Phi(B). \quad (5)$$

$P(B)$ is the prediction of ground truth coordinates and $\Phi(B)$ is the eigenvector composed by the feature maps from the corresponding proposal. In this paper, we adopt the smooth L1 loss function to calculate the distance from the ground truth to the predicted bounding box:

$$\text{Smooth L1} = \begin{cases} 0.5 \cdot (P(B) - G)^2, & \text{if } |P(B) - G| < 1, \\ |P(B) - G| - 0.5, & \text{otherwise.} \end{cases} \quad (6)$$

Therefore, the training optimization objective can be defined as equation (7); here, N is the number of predicted bounding box.

$$\widehat{W} = \arg \min_W \sum_i^N (\text{Smooth L1})_i. \quad (7)$$

From Figure 6, the upper branch of output provides a prediction of the crack shape which is characterized by drawing the predicted mask on the object. More specific,

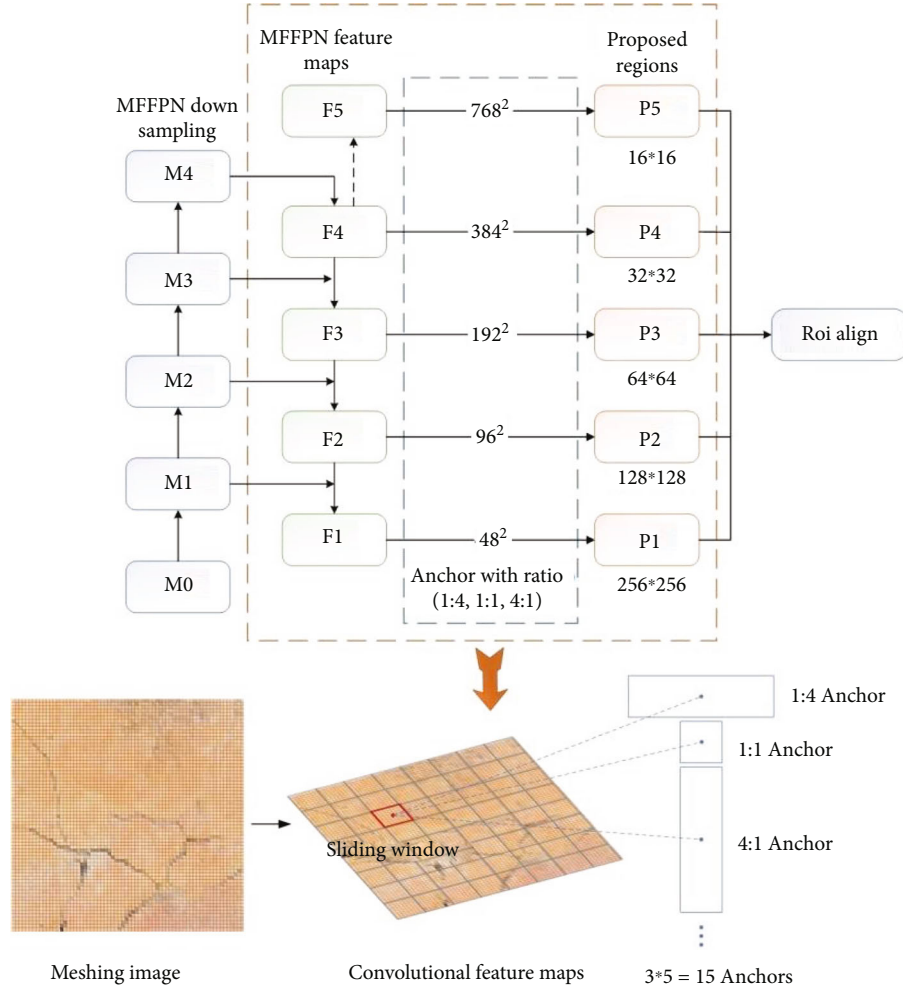


FIGURE 8: Framework of regional proposal network.

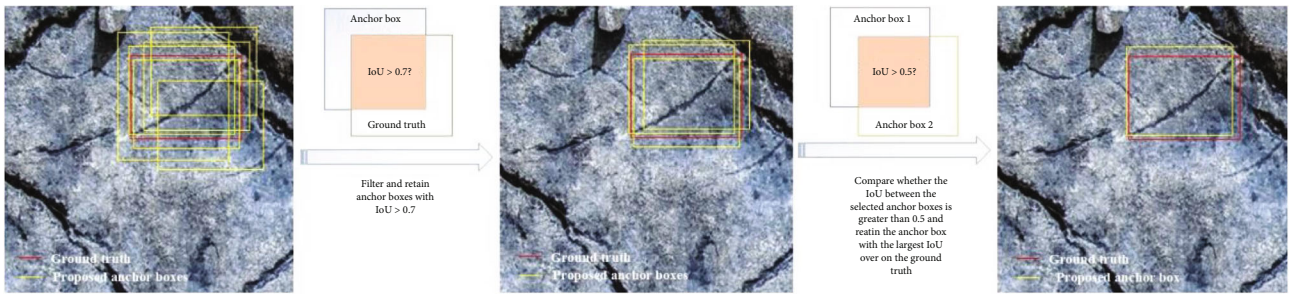


FIGURE 9: Diagram of selecting proper anchors based on the IoU algorithm.

the 7×7 feature maps will be enlarged to 14×14 through a head network to provide robust features to mask prediction. Mask prediction is actually a binary classification calculated by the average binary cross-entropy function as follows:

$$L = \frac{1}{N} \sum_i - [y_i \log(p_i) + (1 - y_i) \cdot \log(1 - p_i)], \quad (8)$$

where y_i is the label of i th proposal with a positive predic-

tion as 1 and a negative prediction as 0. p_i is the predicted probability of cracks of the i th proposal calculated by the sigmoid function.

After the above fine-tuning, the final anchor box will be fed to the crack classification and segmentation network for final outputs.

4.2.4. Loss Function. The detection loss has two parts, which are the loss of the RPN and the loss of the segmentation output. As for the loss function of RPN, it is adopted from faster

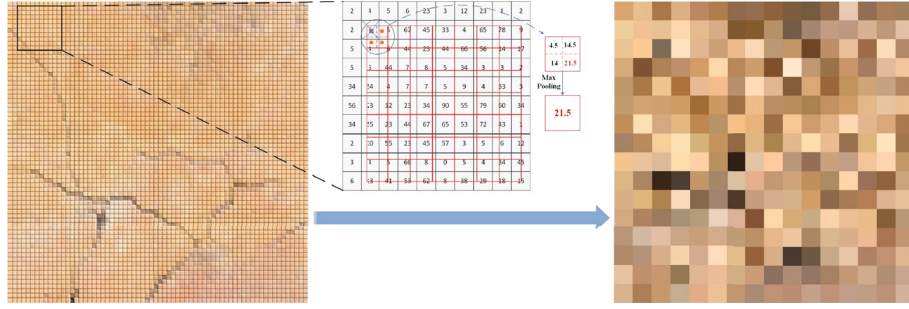


FIGURE 10: Feature map extraction using RoI align.

R-CNN which can be written as

$$L(\{C_i\}, \{P_i\}) = \frac{1}{N_{cls}} \sum_i^N L_{cls}(C_i, C_i^*) + \lambda \frac{1}{N_{reg}} \sum_i^N C_i^* L_{reg}(P_i, G), \tag{9}$$

where C_i is the positive softmax probability of cracks and C_i^* is the IoU of ground truth and C_i , which means only those proposals with IoU larger than 0.7 are considered in the loss function. P_i is the predicted bounding box, G is the ground truth of bounding box, N_{cls} is the number of positive predicted proposal on cracks, and N_{reg} is the number of positive predicted bounding box. L_{cls} is the softmax loss to calculate the predicted loss of cracks on each positive proposal and L_{reg} is the regression loss of bounding box calculated by the smooth $L1$ loss function. λ is the balancing coefficient to balance the difference between N_{cls} and N_{reg} in case that these two values vary too much; in this paper, we simply use $\lambda = N_{reg}/N_{cls}$ to balance the great difference between N_{cls} and N_{reg} if N_{reg}/N_{cls} is over 10.

As for the loss function of the segmentation output, it consists of three parts, namely, the detection class loss, the regression loss of bounding box, and the instance segmentation loss.

$$L_{all} = 0.1 \cdot L_{cls} + 0.6 \cdot L_{reg} + 0.3 \cdot L_{mask}, \tag{10}$$

where L_{cls} and L_{reg} consist with the loss function in RPN and L_{mask} is the average binary cross-entropy function as equation (8). During training, we found that the L_{reg} is hard to train and will have a strong impact on the training results; thus, we manually set the loss weight of L_{reg} larger than the other two components.

4.3. Crack Characterization Method. After successfully segmenting rock cracks via multi-CrackNet, this paper proposed an intelligent quantification method to automatically compute the length and maximum width of each crack. The proposed quantification algorithm flowchart is elucidated in Figure 11. Initially, the segmented cracks are extracted from the background and isolated into individual images that are converted into binary images labelled as 1 and 0, where 1 indicates the crack body and 0 represents the background. Subsequently, a

series of techniques are used to extract the key information and calculate the length and width of the crack.

To illustrate the quantification process clearly, a local demonstration with schematic diagram is provided in Figure 12 to elaborate on 10 steps. To calculate the length and width of the crack, it is very important to extract the main skeleton of the crack right located in the middle. Normally, binarization operation can simplify image data without losing texture information, which leads to easy operation and analysis of skeleton information. After binarization, as indicated in the first step in Figure 12, there exist many irregular spurs around the boundary of the crack, which may significantly affect the performance of skeleton extraction. As demonstrated in the schematic below, at each step, those spurs are denoted as isolated pixels with a value of 1. A simple way to eliminate these isolated pixels is to carry out a patch labelling operation by reassigning different pixel values to all separated pixel patches. As shown in the second step, all isolated pixels will be labelled in different values and distinguished from the main body of the crack. Thus, it is easy to eliminate those non-1 pixels and obtain a spur-free image in the third step. In the fourth step, a traditional distance transform method (DTM) [46] is applied to highlight the position and indicate the distance of the middle line from the edge of the crack.

As demonstrated in the schematic diagram of the fourth step, the number represents the pixel distance of each non-0 pixel to the nearest 0 pixel. In this step, the maximum width of the crack can be easily obtained from the maximum pixel value.

$$\text{width} = \max(\text{pixel value}) \times 2 - 1. \tag{11}$$

To extract the skeleton of the crack, a thinning operation is proposed by corroding the DTM image in step 4; the corrosion operation will keep transforming the edge pixel to 0 until the non-0 pixels can just remain connected as shown in the schematic diagram in step 5. Subsequently, a skeletonized operation is applied to the corroded DTM image to obtain the skeleton of the crack; this can be done by the imbedded MATLAB function “bwmorph”. The length calculation relies on the middle line of the crack, so a branch-cut operation is used to prune unnecessary branches, which can be achieved by the imbedded function “bwmorph” in MATLAB. However, this operation will eliminate the branchpoints and divide the middle line into multiple segments. Thus, the branchpoints

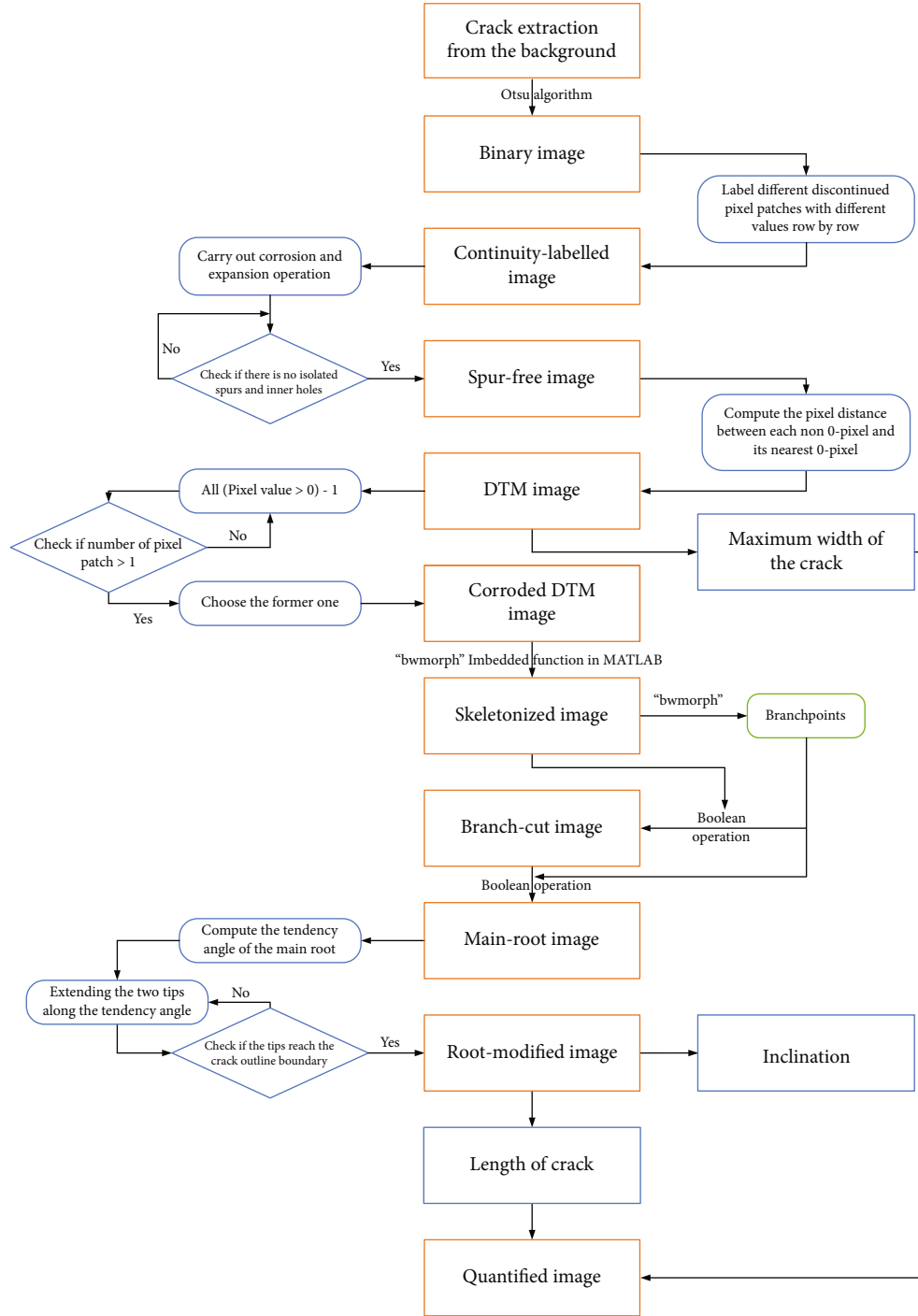


FIGURE 11: Crack quantification algorithm flowchart.

will be added back to maintain the continuity of the main root with Boolean operations. In this step, the length of the crack can be calculated by summing all non-0 pixels, but as indicated in Figure 12, corroded-DTM operation may lead to the loss of pixels at two tips of the crack as shown in step 9. Therefore, in the final step, this paper proposed a modified method by expanding the endpoints of the root to the original edges of the crack along the inclined direction of the crack. The crack inclination is calculated based on the least square method. To be specific, if we assume the fitting line of pixels on the

middle root of the crack as

$$y = ax + b, \quad (12)$$

thus, the goal to compute the most likely slope a is to solve:

$$\min Z = \sum_{i=1}^N [y_i - (a + bx_i)]^2, \quad (13)$$

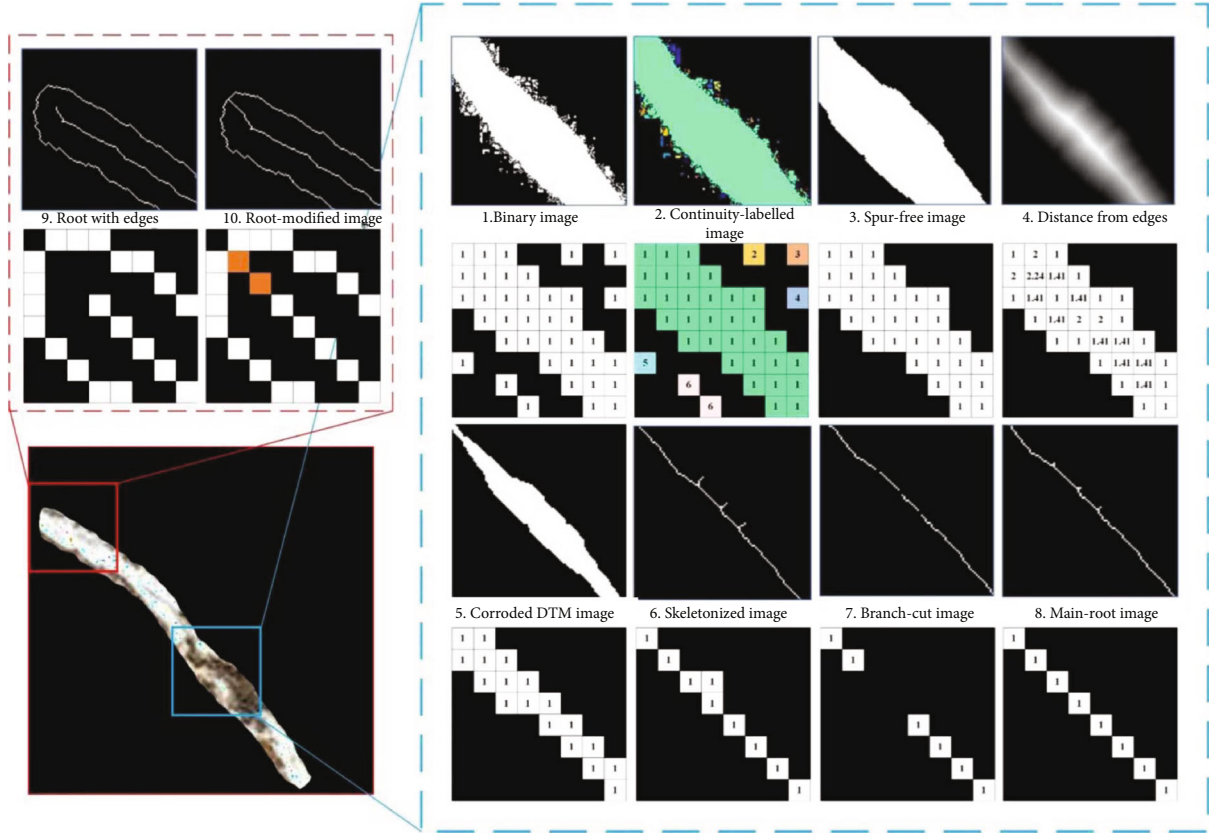


FIGURE 12: Schematic diagram of the proposed quantification algorithm in details.

where $[x_i, y_i]$ is the coordinate of pixels on the middle root. By taking the partial derivative of a and b , the following can be obtained that

$$\frac{\partial}{\partial a} \sum_{i=1}^N [y_i - (a + bx_i)]^2 = -2 \sum_{i=1}^N y_i - (a + bx_i) = 0, \quad (14)$$

$$\frac{\partial}{\partial b} \sum_{i=1}^N [y_i - (a + bx_i)]^2 = -2 \sum_{i=1}^N y_i - (a + bx_i)x_i = 0.$$

Thus, the following equation sets can be derived.

$$\begin{aligned} aN + b \sum x_i &= \sum y_i, \\ a \sum x_i + b \sum x_i^2 &= \sum x_i y_i. \end{aligned} \quad (15)$$

By solving (15), the slope of the fitting line can be obtained as

$$\hat{a} = \frac{(\sum x_i^2)(\sum y_i) - (\sum x_i)(\sum x_i y_i)}{N \sum x_i^2 - (\sum x_i)^2}. \quad (16)$$

Therefore, the inclination of the crack can be calculated as follows: the fitting line of the crack is elucidated in

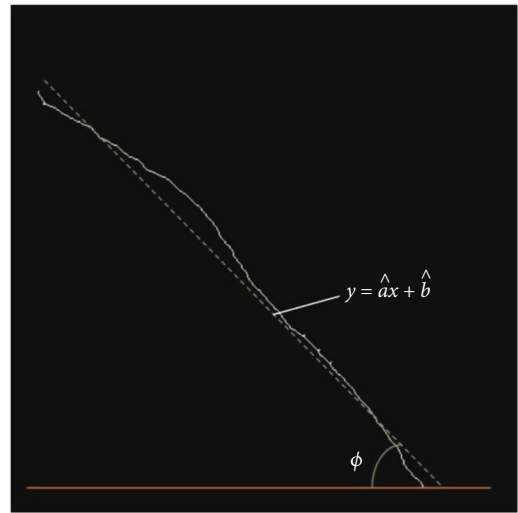


FIGURE 13: Fitting line of the crack.

Figure 13.

$$\phi = \text{actan}(\hat{a}). \quad (17)$$

Since now, the actual length, the inclination, and the maximum width of the crack can be obtained and displayed as shown in Figure 4.

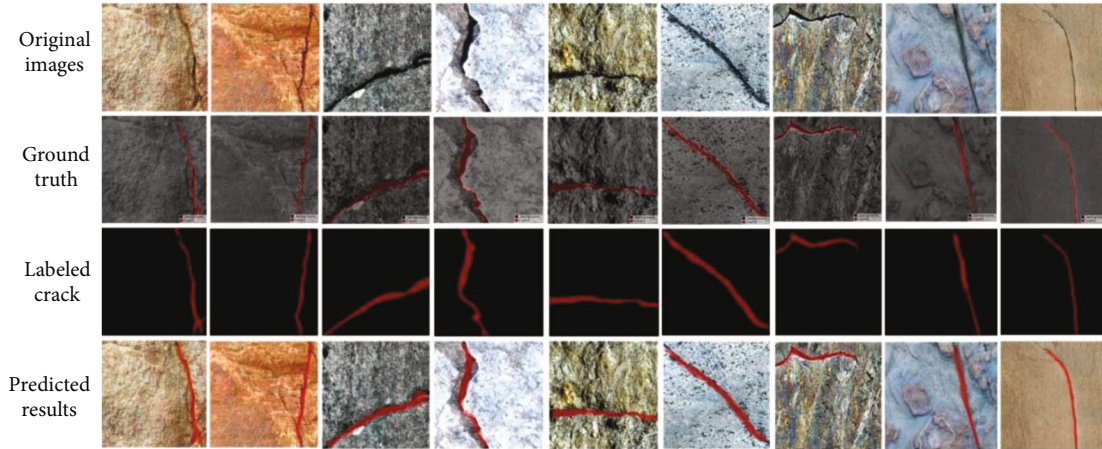


FIGURE 14: Examples of predicted results of the multi-CrackNet on single-crack detection.

5. Case Study and Discussion

In this study, multi-CrackNet was first used to train with captured image dataset from Anshan West in Liaoning Province, China. Images captured in this area contain a variety of rock types, for instance, sandstone, granite stone, gneiss, and limestone. 10% of images are randomly extracted from the datasets to validate and test our training results. After segmentation, segmented images with individual cracks are fed to the proposed quantification algorithm to automatically display the length and width of the crack at the pixel level. The details are elaborated as follows.

5.1. Validation of the Multi-CrackNet. The proposed multi-CrackNet framework is implemented using Python 3.6.3 in Keras environment with the help of Nvidia GeForce RTX 3080 for training. The algorithm runs on an AMD Ryzen 3900X@3.8GHz 12 core processor with 64GB RAM on a Windows 10 PC.

In this experiment, we trained the model in three stages as indicated in Figure 4. Initially, the primary database S0 is cropped to form the secondary database S1 which is then divided into 3 subdatabases according to the number of cracks on the rock. The model training starts with the subdatabase S1-1 which contains only a single crack on the rock. Figure 14 elucidates some examples of the results of the multi-CrackNet on single-crack detection; the result demonstrates robust and confident performance of the method on the segmentation of single crack on different types of rocks/background.

In the second stage, the well-trained model on single-crack detection is applied to train the subdatabase S1-2 in which those images with 2-3 sets of cracks. Figure 15 shows robust predicted results on some rock images across different backgrounds.

At this stage, even though the model has demonstrated great permanence on crack segmentation to some degree, the model makes several mistakes when facing complex backgrounds; Figure 16 depicts several situations when the model may provide false segmentation results. Situa-

tions in Figures 16(a), 16(b), and 16(d) normally happen at the edge of the image, especially when an isolated tiny crack or part of a crack lies at the edge of the image and is contiguous to other cracks. Situation in Figure 16(c) normally occurs in high-contrast images with many tiny and thin cracks that are omitted to be labelled. Situation in Figure 16(e) occurs in low-contrast images with many tiny and thin cracks which may be overdetected by the model.

To overcome the above problem, we continue to apply the fine-tuned model from the second stage to the subdatabase S1-3 and keep improving the performance of the model on the images with more than 3 sets of cracks. Figure 17 shows the predicted results of some examples. As demonstrated in Figure 17, after three-stage training, the model has proved to be relatively reliable and confident for the segmentation of multiple cracks on complex backgrounds.

After three-stage training, the model is capable of handling 1-9 sets of cracks with relatively lower loss, which succeeds in generalizing to crack detection and segmentation for more complex situations with multiple sets of cracks. Figure 18 shows the training and fine-tuning accuracy of three-stage training, where training accuracy indicates the accuracy of head network training results based on transfer learning.

As demonstrated, in the first stage, we train the model with single-crack images for 50 epochs and spend 190 epochs fine-tuning the model until it converges; the segmentation accuracy finally stabilizes at around 89.3%. In the second stage, the model spends 100 epochs on training and 500 epochs on fine-tuning, converging to an accuracy of 83.5%. In the third stage, we used 491 epochs and 508 epochs to train and fine-tune the model, respectively, achieving a total accuracy of 87.1%. It is worth noting that in the first two stages, fine-tuning can greatly improve the performance of the model by quickly decreasing the segmentation loss, but in the third stage, fine-tuning accuracy is pretty close to the training accuracy. This can be simply explained by that the model has already learned enough from the former stages, and the diagram also shows that the two fine-tuning loss curves of stages 2 and 3 nearly converge together, which

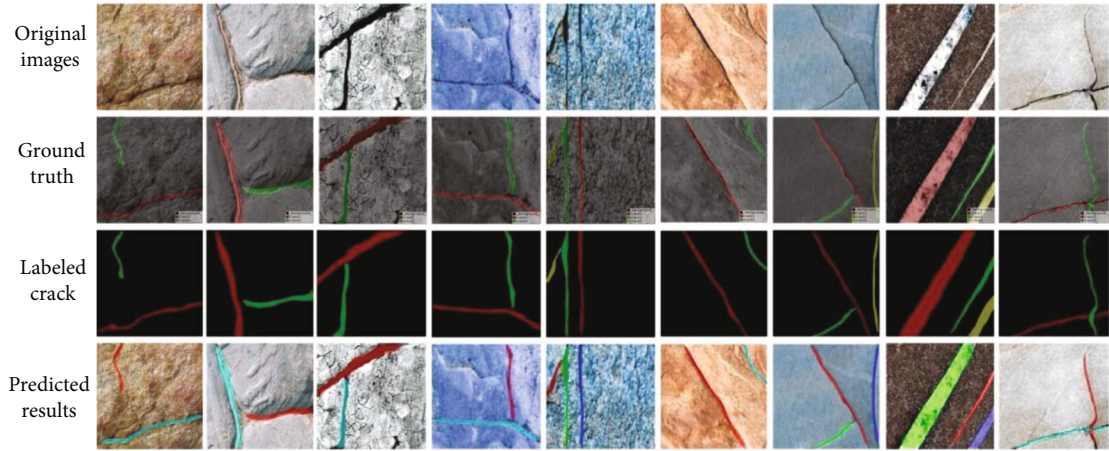


FIGURE 15: Examples of predicted results of the multi-CrackNet on 2-3 sets of crack detections.

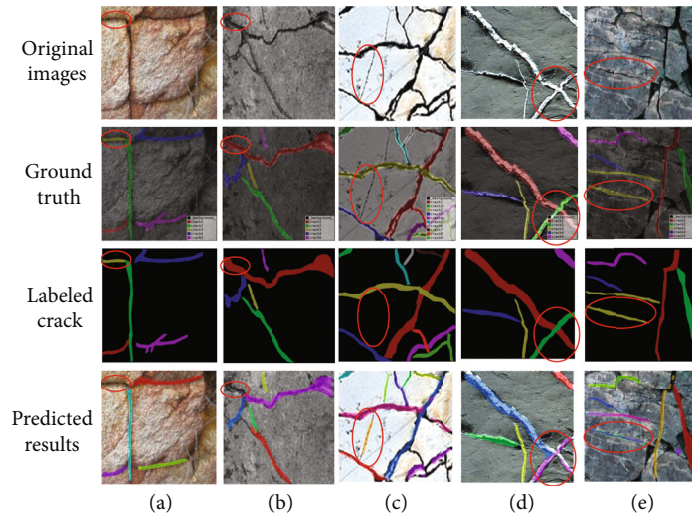


FIGURE 16: Examples of falsely detected situations indicated in red circles. (a) Omitting of crack detection at the edge of the image; (b) missing part of the crack at the edge of the image; (c) providing potential crack detections that are not labelled in the ground truth; (d) inaccurate segmentation when overlapping happens at the edge of the images; and (e) providing redundant predictions on the same crack.

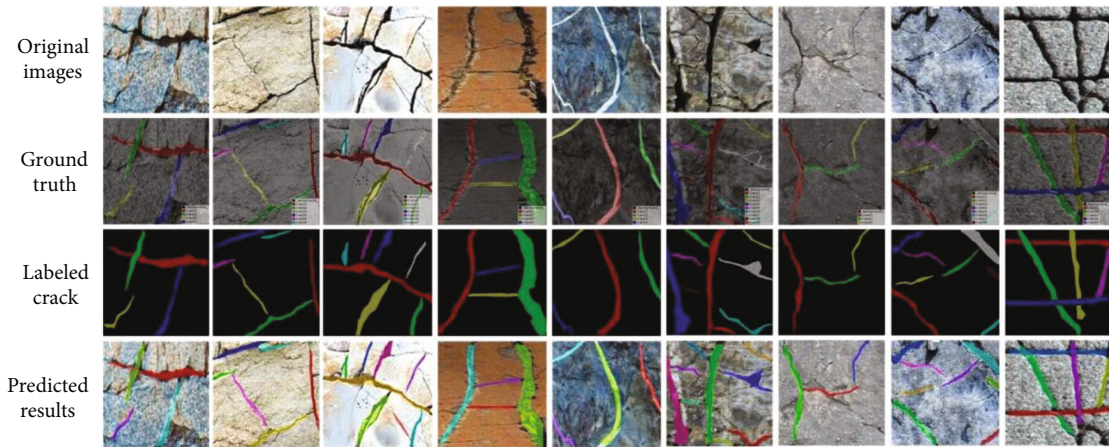


FIGURE 17: Examples of predicted results of the multi-CrackNet on 3 more sets of crack detections.

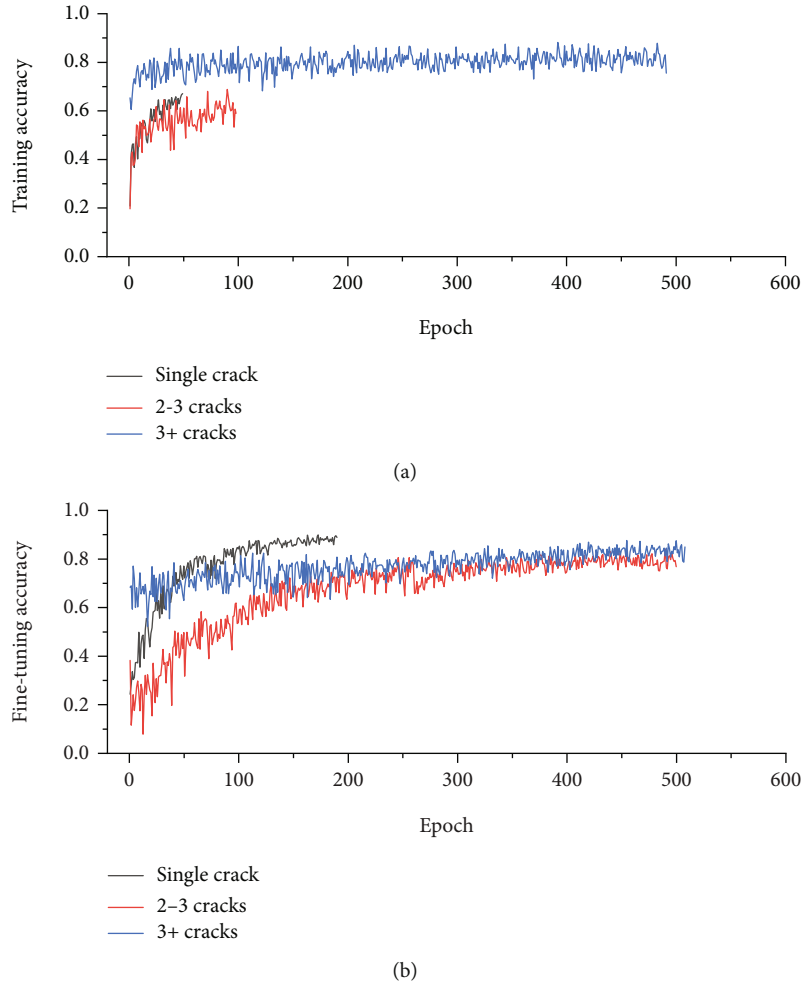


FIGURE 18: Training and fine-tuning process. (a) Training accuracy; (b) fine-tuning accuracy. (Single crack represents the model only trains on subdatabase S1-1 in the first stage; 2-3 cracks represent the model trains on subdatabases S1-1 and S1-2 in the second stage; 3+ cracks represent the model trains on subdatabases S1-1, S1-2, and S1-3 in the third stage).

indicates that the third stage of training is designed to improve the results of the first two stages. Figures 16 and 17 also reveal the same conclusion that the model is able to handle multiple-crack segmentation after training on subdatabases S1-1 and S1-2 to some extent; only some minor mistakes occur when tackling staggered crack segmentation at the edge of the images. Training in the third stage aims at improving the robustness and reliability of the model to generalize the segmentation ability to more cases.

To compare the performance of the method to that of the current deep-learning-based method, faster R-CNN and mask R-CNN are selected. The identical training databases used in this paper are utilized to train Unet++ and mask R-CNN. Figure 19 shows some examples of the comparative results among the methods, Unet++ and mask R-CNN. It is apparent that current methods can barely handle such complex segmentation situations when multiple cracks are staggering with each other, but our method provides robust and accurate predicted results across different types of complex background. It is noticeable that both Unet++ and mask R-CNN can localize the approximate location of each crack, but the point is that they can-

not provide precise segmentation for cracks. Specifically, by replacing FPN with MFFPN, we can see a great improvement on crack segmentation performance. This may be explained by the fact that the current model cannot extract enough hybrid features from FPN to identify complex layouts of cracks and finally segment them, respectively.

5.2. Validation of the Quantification Method. The proposed multi-CrackNet provides robust and accurate segmentation results of rock cracks on different rock surfaces. These segmented cracks are then processed by a new quantification method based on DTM to measure the length and width of each crack. Traditional measuring algorithms such as the fast march propagation method [48] are applicable for thin crack only without considering the width of the crack; our method used hybrid techniques to compute the length and width of the crack regardless of the shape of the crack. Some report methods also considered pixel-level quantification of cracks, but they neglected the general inclined tendency of the crack in filling the errors between the tips and the edges of the crack [49]. Our DTM-based method considers the

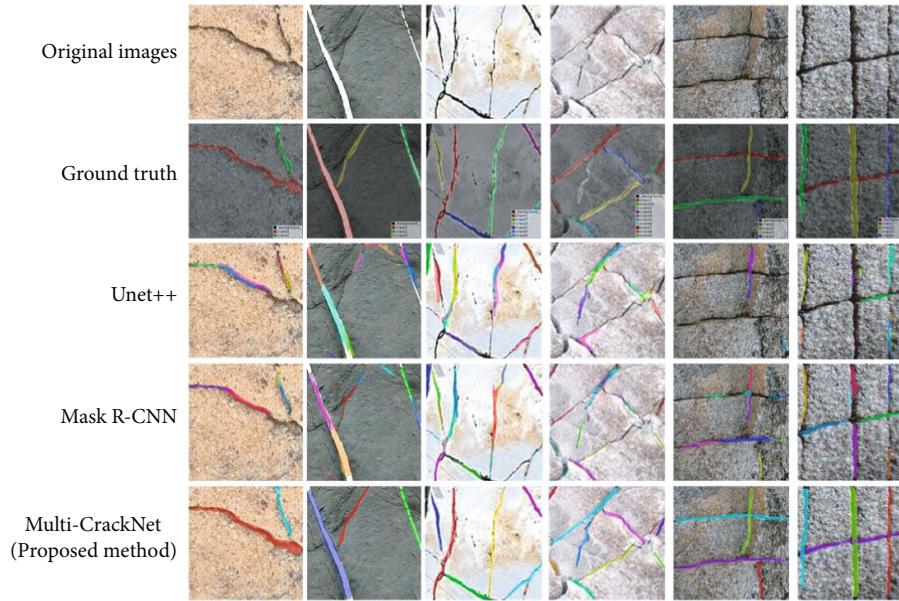


FIGURE 19: Comparative studies among multi-CrackNet, Unet++ [47], and mask R-CNN.

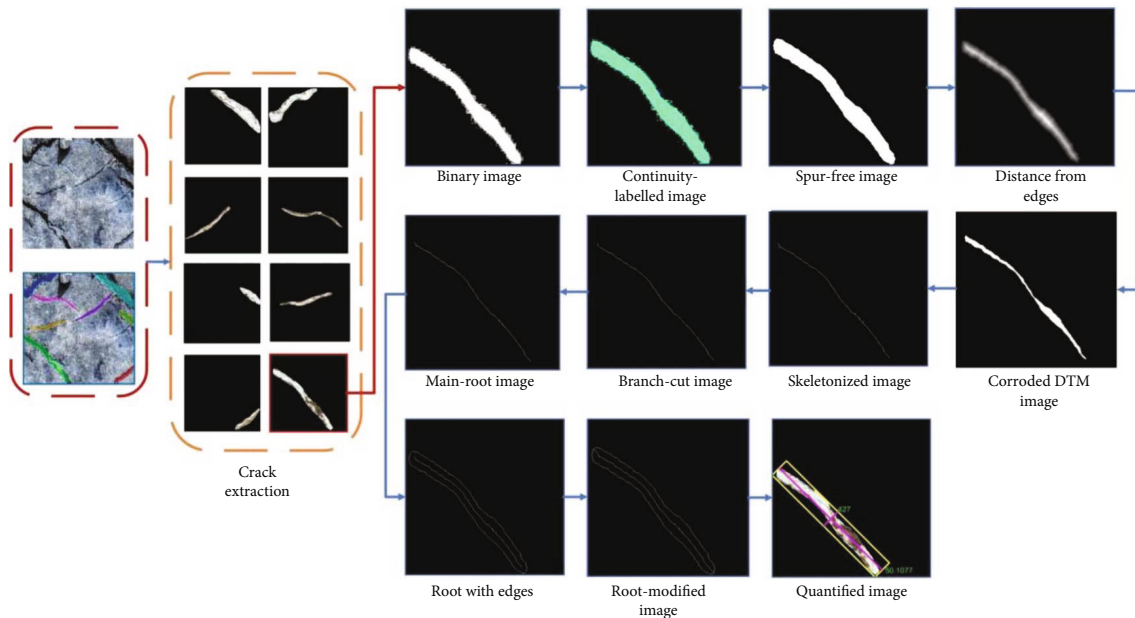


FIGURE 20: Operating process of the proposed quantification algorithm.

tendency angle of each crack and applies an extension operation on the two tips by lengthening the pixels at tips to the edges in a rational direction.

Figure 20 demonstrates the operating process of the proposed quantification method; as elucidated, the goal of this quantification algorithm is to precisely extract the central line of the crack and the distance from the central line to the edges. Theoretically, this algorithm can accurately attain the length and width of crack from the segmented images because the algorithm can eliminate noise pixels and merge all pixels along the central line from which the geometry information can be derived.

To verify the performance of the quantification results, some examples are presented in Figure 21. Results show that our method can handle different types of crack shape with multiple scales. It is noticeable that the proposed algorithm can properly measure the crack length with an accuracy of single digit but measure the crack width in decimals. This is because the measurement for crack length is to simply sum up the pixels in the central line, while the measurement for crack width is to compute the distance from the central line to the edges; that is, the measured distance may not be integer. This means the measurement in width is more accurate than that in length. However, the quantification method is limited by computing the

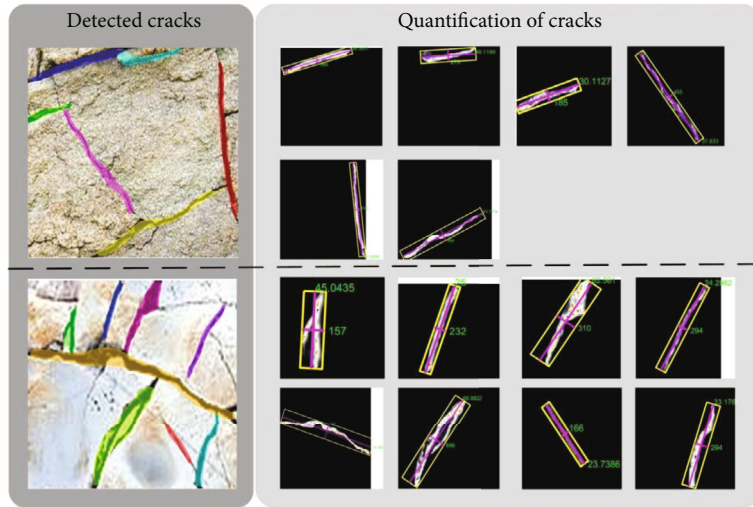


FIGURE 21: Example results of the proposed method for crack quantification.

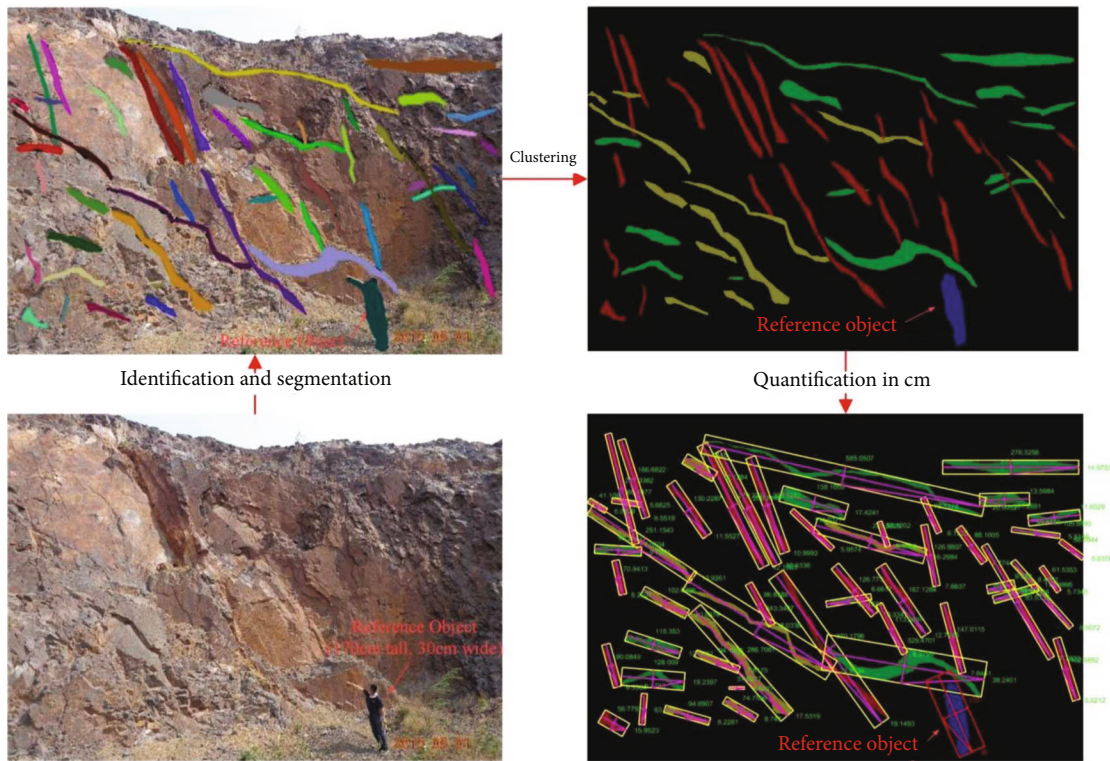


FIGURE 22: Quantification in cm of discontinuity trace on the selected outcrop with a reference object.

geometry information in pixels but not in real-world units, such as mm. This limitation can be offset by knowing the distance between the rock face and the camera lens or in other way, for instance, a reference object with known geometry information is also captured in the same images.

5.3. Case Study of a Selective Outcrop. To investigate the performance of our proposed method in macroscale, a selected outcrop from Anshan West accompanied with a reference person with 170 cm height and 30 cm shoulder breadth

within the selected sample window is captured by a smart-phone (iPhone X @1200 megapixel). The multi-CrackNet initially cropped the original image into 16 subimages and identified each crack on each subimage. These subimages with identified cracks were then merged together to form 51 macrolinear discontinuities; it is noticeable that the reference person is also detected due to transferred learning from COCO dataset. To understand the relationship of discontinuities in this outcrop, a clustering operation was applied to indicate the different sets of discontinuities.

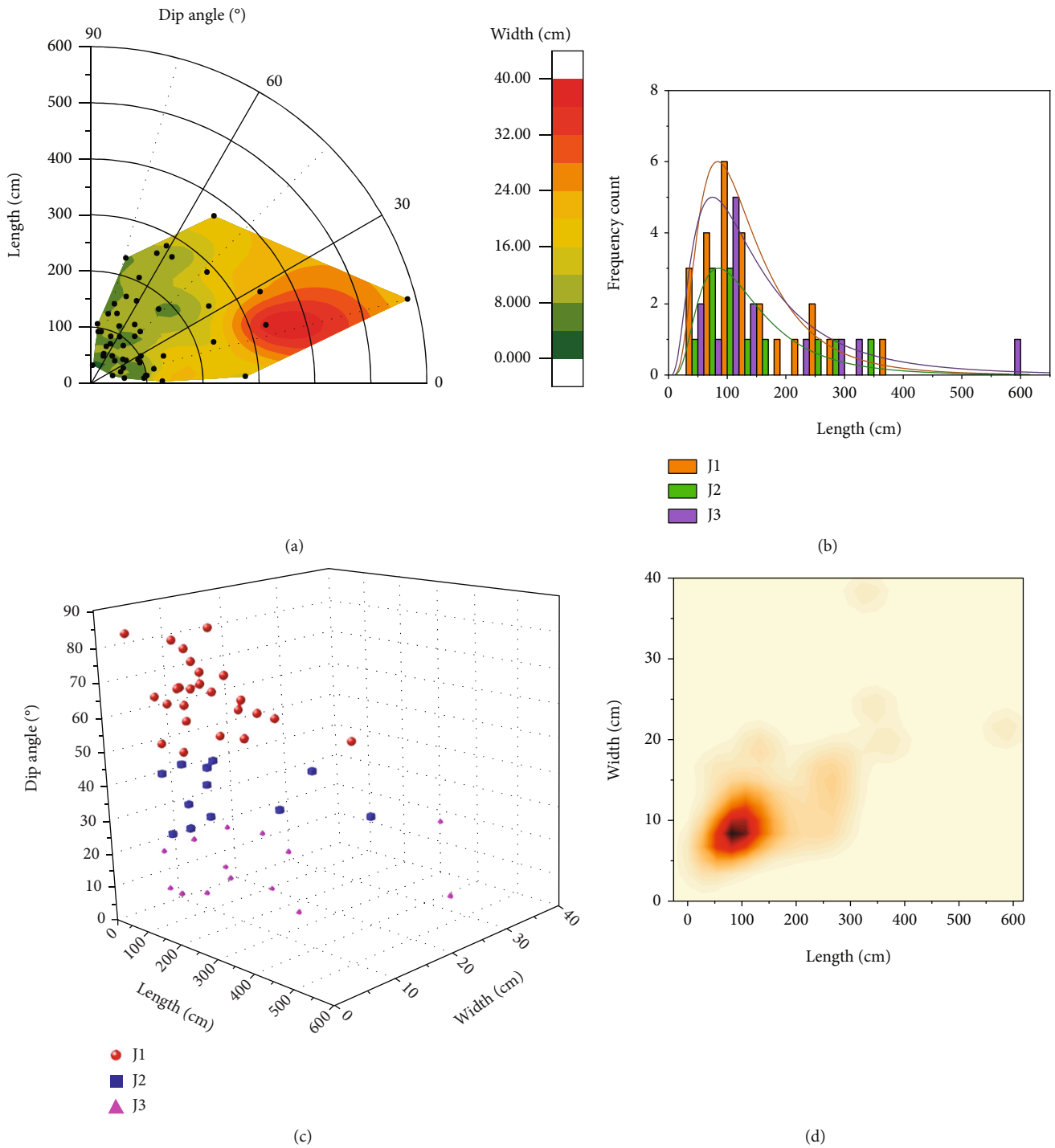


FIGURE 23: Statistical parameters of the linear discontinuities. (a) Dip angle contour plot considering length and width. (b) Frequency count of 3 sets of linear discontinuities. (c) Clustering plot of 3 sets of linear discontinuities. (d) Geometrical size distribution of discontinuities.

As shown in Figure 22, there are 3 sets of discontinuities labelled in red (J1), green (J2), and yellow (J3); the reference person was denoised and labelled in blue. At last, the proposed quantification algorithm was used to compute the length and width of each linear discontinuity in cm by converting the pixel value to centimeters with the help of the reference person. As demonstrated, the reference person was quantified in cm precisely and highlight in a red rectan-

gular box. The rectangular boxes in Figure 22 do not indicate the length and width of the detected objects; the boxes only indicate the position of each detected objects.

According to the quantification results, the statistical parameters of the identified discontinuities are shown in Figure 23. Based on the dominance dip angle, these linear discontinuities can be classified into 3 sets denoted as J1, J2, and J3, which are distinctive in average dip angle

TABLE 2: Geometry parameters obtained by the proposed method.

Discontinuity set	Number	Manual investigation			Our method			Differences		
		Ave L (cm)	Ave W (cm)	Ave dip ang ($^{\circ}$)	Ave L (cm)	Ave W (cm)	Ave dip ang ($^{\circ}$)	$\Delta dl $	$\Delta DW $	$\Delta DD $
J1	25	138.3	9.1	69.2	134.899	8.706	66.32	3.401	0.394	2.88
J2	12	150.2	11.2	34.2	147.099	10.819	36.24	3.101	0.381	2.04
J3	14	170.5	15.1	13.5	168.042	14.567	12.71	2.458	0.533	0.79

* $\Delta|DL|$, $\Delta|DW|$, and $\Delta|DD|$ represent the differences in average length, average width, and average dip angle, respectively.

computed in Table 2. The statistical data shows that J1 dominate the majority with an average dip angle of 69.2° which is align with the site investigation in Section 3. Further information can be read from Figure 23(b) that the trace length of most discontinuities lies in the range of 50 to 200 cm and the width of most discontinuities does not exceed 15 cm, which can be seen from Figure 23(d) that the trace length of 3 sets of discontinuities also follow a good lognormal distribution. To validate our method in geometry quantification of discontinuities, Table 2 also shows the statistical data from manual measurement onsite; the difference between the manual investigation and the proposed method are relatively small, which means our hybrid method is convincing to some extent.

6. Conclusions

This paper developed a fine-grained segmentation model called multi-CrackNet based on the advantages of mask R-CNN to segment objects at pixel level. To realize the general segmentation ability of linear discontinuities to various types of background in different conditions, the following contributions were achieved:

- (1) A multiscale feature fusion pyramid network (MFFPN) was developed based on FPN to capture more sensitive features of cracks across different types of rock. Compared with current feature extraction techniques, the method can extract and fuse more scales of features across different complex backgrounds or conditions. Experiments show that the method can achieve a prediction accuracy of 87.1% for 1-9 sets of cracks over different types of rock and provide robust and reliable segmentation results onsite
- (2) A novel operation scheme was proposed by dividing the training tasks into 3 stages to simulate the human-based learning process, starting with the easy task (one set of crack) and then enhancing segmentation ability by feeding on advanced tasks (three more sets of cracks). This training scheme can greatly contribute to the robust performance of challenging identification and segmentation tasks, especially targeting segmentation tasks with extremely complex layouts or backgrounds
- (3) The well-designed algorithm for crack quantification combines hybrid techniques to compute the length and width of the crack in pixel level. The method sees great potential for onsite investigation because

it can automatically identify and quantify each crack by simply taking a picture of the rock surface, which reduces a large amount of labor force. A case study in Anshan West was performed to validate the reliability of our hybrid method in identifying and quantifying engineering scale linear discontinuities; result shows that our hybrid method can accurately identify the trace line of each discontinuity and compute their length, width, and dip angle

However, due to limited information from a single image, we cannot outperform the current 3D-cloud-point-based methods that can acquire more geometry information such as strike and dip; our method shows great handiness and portability for those sitework that simply want to know the very basic parameters of the outcrops.

Data Availability

The data presented in this study are available on request from the corresponding author.

Conflicts of Interest

The authors declare that they have no conflict of interest.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (52274107), the Interdisciplinary Research Project for Young Teachers of University of Science and Technology Beijing (FRF-IDRY-GD21-001), and the Foshan Science and Technology Innovation Special Fund Funding Project (BK20BE008).

References

- [1] S. Slob, B. van Knapen, R. Hack, K. Turner, and J. Kemeny, "Method for automated discontinuity analysis of rock slopes with three-dimensional laser scanning," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1913, no. 1, pp. 187–194, 2005.
- [2] J. Zhou, X. B. Li, and H. S. Mitri, "Evaluation method of rockburst: state-of-the-art literature review," *Tunnelling and Underground Space Technology*, vol. 81, pp. 632–659, 2018.
- [3] J. Zhou, A. Nekouie, C. A. Arslan, B. T. Pham, and M. Hasanipanah, "Novel approach for forecasting the blast-induced AOp using a hybrid fuzzy system and firefly algorithm," *Engineering with Computers*, vol. 36, no. 2, pp. 703–712, 2020.

- [4] P. Assali, P. Grussenmeyer, T. Villemin, N. Pollet, and F. Viguier, "Solid images for geostructural mapping and key block modeling of rock discontinuities," *Computers & Geosciences*, vol. 89, pp. 21–31, 2016.
- [5] D. Giordan, Y. Hayakawa, F. Nex, F. Remondino, and P. Tarolli, "Review article: the use of remotely piloted aircraft systems (RPASs) for natural hazards monitoring and management," *Natural Hazards and Earth System Sciences*, vol. 18, no. 4, pp. 1079–1096, 2018.
- [6] W. C. Haneberg, "Using close range terrestrial digital photogrammetry for 3-D rock slope modeling and discontinuity mapping in the United States," *Bulletin of Engineering Geology and the Environment*, vol. 67, no. 4, pp. 457–469, 2008.
- [7] D. Zekkos, W. Greenwood, J. Lynch et al., "Lessons learned from the application of UAV-enabled structure-from-motion photogrammetry in geotechnical engineering," *International Journal of Geoenvironmental Case Histories*, vol. 4, no. 4, pp. 254–274, 2018.
- [8] J. Chen, H. Zhu, and X. Li, "Automatic extraction of discontinuity orientation from rock mass surface 3D point cloud," *Computers & Geosciences*, vol. 95, pp. 18–31, 2016.
- [9] D. Deb, S. Hariharan, U. M. Rao, and C.-H. Ryu, "Automatic detection and analysis of discontinuity geometry of rock mass from digital images," *Computers & Geosciences*, vol. 34, no. 2, pp. 115–126, 2008.
- [10] A. J. Riquelme, A. Abellán, R. Tomás, and M. Jaboyedoff, "A new approach for semi-automatic rock mass joints recognition from 3D point clouds," *Computers & Geosciences*, vol. 68, pp. 38–52, 2014.
- [11] M. Sturzenegger, D. Stead, and D. Elmo, "Terrestrial remote sensing-based estimation of mean trace length, trace intensity and block size/shape," *Engineering Geology*, vol. 119, no. 3–4, pp. 96–111, 2011.
- [12] N. Barton, "Suggested methods for the quantitative description of discontinuities in rock masses," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, vol. 15, no. 6, pp. 319–368, 1978.
- [13] J. Muralha, G. Grasselli, B. Tatone, M. Blümel, P. Chryssanthakis, and J. Yuqing, "ISRM suggested method for laboratory determination of the shear strength of rock joints: revised version," *Rock Mechanics and Rock Engineering*, vol. 47, no. 1, pp. 291–302, 2013.
- [14] X. Li, J. Chen, and H. Zhu, "A new method for automated discontinuity trace mapping on rock mass 3D surface model," *Computers & Geosciences*, vol. 89, pp. 118–131, 2016.
- [15] J. Guo, L. Wu, M. Zhang, S. Liu, and X. Sun, "Towards automatic discontinuity trace extraction from rock mass point cloud without triangulation," *International Journal of Rock Mechanics and Mining Sciences*, vol. 112, pp. 226–237, 2018.
- [16] D. Kong, F. Wu, and C. Saroglou, "Automatic identification and characterization of discontinuities in rock masses from 3D point clouds," *Engineering Geology*, vol. 265, article 105442, 2020.
- [17] F.-C. Chen and M. R. Jahanshahi, "NB-CNN: deep learning-based crack detection using convolutional neural network and Naïve Bayes data fusion," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4392–4400, 2018.
- [18] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," *Automation in Construction*, vol. 99, pp. 52–58, 2019.
- [19] D. H. Kang and Y. J. Cha, "Autonomous UAVs for structural health monitoring using deep learning and an ultrasonic Beacon system with geo-tagging," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 10, pp. 885–902, 2018.
- [20] Z. Qu, F. R. Ju, Y. Guo, L. Bai, and K. Chen, "Concrete surface crack detection with the improved pre-extraction and the second percolation processing methods," *PLoS One*, vol. 13, no. 7, article e0201109, 2018.
- [21] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyükoztürk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 731–747, 2018.
- [22] Y. J. Cha, K. You, and W. Choi, "Vision-based detection of loosened bolts using the Hough transform and support vector machines," *Automation in Construction*, vol. 71, pp. 181–188, 2016.
- [23] J. C. P. Cheng and M. Z. Wang, "Automated detection of sewer pipe defects in closed-circuit television images using deep learning techniques," *Automation in Construction*, vol. 95, pp. 155–171, 2018.
- [24] S. S. Kumar, D. M. Abraham, M. R. Jahanshahi, T. Iseley, and J. Starr, "Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks," *Automation in Construction*, vol. 91, pp. 273–283, 2018.
- [25] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.
- [26] F. G. Praticò, R. Fedele, V. Naumov, and T. Sauer, "Detection and monitoring of bottom-up cracks in road pavement using a machine-learning approach," *Algorithms*, vol. 13, no. 4, p. 81, 2020.
- [27] J. Chen, H. Huang, A. G. Cohn, D. Zhang, and M. Zhou, "Machine learning-based classification of rock discontinuity trace: SMOTE oversampling integrated with GBT ensemble learning," *International Journal of Mining Science and Technology*, vol. 32, no. 2, pp. 309–322, 2022.
- [28] J. Chen, D. Zhang, H. Huang, M. Shadabfar, M. Zhou, and T. Yang, "Image-based segmentation and quantification of weak interlayers in rock tunnel face via deep learning," *Automation in Construction*, vol. 120, article 103371, 2020.
- [29] J. Chen, M. Zhou, H. Huang, D. Zhang, and Z. Peng, "Automated extraction and evaluation of fracture trace maps from rock tunnel face images via deep learning," *International Journal of Rock Mechanics and Mining Sciences*, vol. 142, article 104745, 2021.
- [30] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, Venice, Italy, 2017.
- [31] Z. Ren, Y. Tan, L. Huang, and H. Yu, "Optimization of automatic extraction procedure for particles in asphalt mixture towards superior robustness and accuracy," *Construction and Building Materials*, vol. 342, article 128002, 2022.
- [32] Y. Tang, L. He, W. Lu, X. Huang, H. Wei, and H. Xiao, "A novel approach for fracture skeleton extraction from rock surface images," *International Journal of Rock Mechanics and Mining Sciences*, vol. 142, article 104732, 2021.

- [33] C. Huang, X. Li, and Y. Wen, "An OTSU image segmentation based on fruitfly optimization algorithm," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 183–188, 2021.
- [34] Y. Piao, Y. Jiang, M. Zhang, J. Wang, and H. Lu, "PANet: patch-aware network for light field salient object detection," *IEEE Transactions on Cybernetics*, 2021.
- [35] A. Isar, C. Nafornta, and G. Magu, "Hyperanalytic wavelet-based robust edge detection," *Remote Sensing*, vol. 13, no. 15, p. 2888, 2021.
- [36] F. Liu, J. Liu, and L. Wang, "Asphalt pavement crack detection based on convolutional neural network and infrared thermography," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22145–22155, 2022.
- [37] X. Wu and X. Liu, "Building crack identification and total quality management method based on deep learning," *Pattern Recognition Letters*, vol. 145, pp. 225–231, 2021.
- [38] E. Ibragimov, H.-J. Lee, J.-J. Lee, and N. Kim, "Automated pavement distress detection using region based convolutional neural networks," *International Journal of Pavement Engineering*, vol. 23, no. 6, pp. 1981–1992, 2022.
- [39] L. Attard, C. J. Debono, G. Valentino, M. D. Castro, A. Masi, and L. Scibile, "Automatic crack detection using mask R-CNN," in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 152–157, Dubrovnik, Croatia, Sept. 2019.
- [40] H. L. Ramandi, S. Irtza, T. Sirojan et al., "FracDetect: a novel algorithm for 3D fracture detection in digital fractured rocks," *Journal of Hydrology*, vol. 607, article 127482, 2022.
- [41] D. Ai, G. Jiang, S.-K. Lam, P. He, and C. Li, "Automatic pixel-wise detection of evolving cracks on rock surface in video data," *Automation in Construction*, vol. 119, article 103378, 2020.
- [42] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: a database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 157–173, 2008.
- [43] T.-Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8693 of Lecture Notes in Computer Science, , pp. 740–755, Springer, Cham, 2014.
- [44] A. Neubeck and L. van Gool, "Efficient non-maximum suppression," in *18th International Conference on Pattern Recognition (ICPR'06)*, pp. 850–855, Hong Kong, China, 2006.
- [45] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," 2016, <http://arxiv.org/abs/1612.02295>.
- [46] D. W. Paglieroni, "Distance transforms: properties and machine vision applications," *CVGIP: Graphical Models and Image Processing*, vol. 54, no. 1, pp. 56–74, 1992.
- [47] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: a nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Springer, Cham, 2018.
- [48] Y. J. Tsai, V. Kaul, and A. Yezzi, "Automating the crack map detection process for machine operated crack sealer," *Automation in Construction*, vol. 31, pp. 10–18, 2013.
- [49] L. Y. Ding, W. L. Fang, H. B. Luo, P. E. D. Love, B. T. Zhong, and X. Ouyang, "A deep hybrid learning model to detect unsafe behavior: integrating convolution neural networks and long short-term memory," *Automation in Construction*, vol. 86, pp. 118–124, 2018.