



Automated extraction and evaluation of fracture trace maps from rock tunnel face images via deep learning

Jiayao Chen^a, Mingliang Zhou^{a,*}, Hongwei Huang^a, Dongming Zhang^a, Zhicheng Peng^b

^a Key Laboratory of Geotechnical and Underground Engineering of Ministry of Education and Department of Geotechnical Engineering, Tongji University, Shanghai, China
^b Guanghua School of Management, Peking University, Beijing, China

ARTICLE INFO

Keywords:

Fracture trace map
Rock tunnel face
Convolutional neural network
Computer vision
Fracture evaluation

ABSTRACT

This paper proposes an image-based method for automated rock fracture segmentation and fracture trace quantification. It is integrated using a CNN-based model named FraSegNet, a skeleton extraction algorithm, and a chain code-based polyline approximation algorithm. A rock tunnel fracture database with a total of 3,000 images of rock tunnel faces is established and selected to train and test the FraSegNet model. A comparison study is further conducted and shows that the FraSegNet model shows advanced performance in pixel-level fracture trace map extraction and noise reduction compared to other deep learning approaches and traditional image edge detection algorithms. Next, the skeletons of the predicted fracture trace maps are extracted and the corresponding polyline for each fracture skeleton is thus obtained and output as a text file composed of key nodes coordinates. The fracture trace characteristics (trace length, dip angle, density, and intensity) are acquired using node-based files. The quantitative evaluation of the proposed method illustrates that it can extract trace occurrences effectively and accurately. A case study of three full scale tunnel sections demonstrates the proposed method to be an efficient approach for acquiring and evaluating 2D fracture occurrences of under-construction rock tunnel faces.

1. Introduction

Grasping the quantitative characteristics of fracture traces in a rock tunnel face is of great significance for determining construction arrangements for tunnel excavation and supporting measures.^{1,2} These characteristics, consisting of trace length, trace dip angle, trace density, and trace intensity,^{3–5} are often evaluated from fracture trace maps obtained from visual inspection of the tunnel faces. Traditionally, fracture trace maps of rock tunnel faces are sketched by the field engineers, where they have to bear safety risks instigated by close observation. This manual inspection process is a labour- and time-consuming task, and is associated with large bias or even error due to unfavourable site conditions.^{6,7} Hence, there is an urgent demand to explore a computer vision-based inspection approach that can automatically extract and evaluate fracture trace information in rock tunnel faces.

Feature-based extraction algorithms have been commonly used over the last twenty years^{8–15} to produce fracture trace maps from digital images of rock mass exposure. These techniques can capture pixel- or grid-level features with satisfactory results by applying various pixel

matrix algorithms (e.g. threshold segmentation, image enhancement, edge detection, etc.).^{13,14,16–20} However, such feature extraction methods have brought obstacles to field engineers and constructors since each method is proposed for a specific application scene. Once the application scene changes, the corresponding method may show poor performance.^{21–24} Moreover, the feature extraction ability of these algorithms for detecting target objects is focused on the pixel level rather than the semantics level which can extract the target features from the multi-layers of an image.^{25,26}

With the fast growing research advances in the computer vision domain, automatic feature extraction using deep learning algorithms can perform better in comparison with the aforementioned feature-based methods. Among the different branches of deep learning-based methods, convolutional neural network (CNN)-based feature extraction models have shown superior performances in many inspection tasks.^{27–30} The CNN-based models can be exploited to acquire high-level features and are then integrated with a classifier to achieve complex feature recognition tasks. To date, researchers have proposed various CNN-based models (FCN,³¹ DeepLab,^{32,33} U-Net,³⁴ SegNet,³⁵ etc.), the

* Corresponding author.

E-mail addresses: 1810181@tongji.edu.cn (J. Chen), zhoum@tongji.edu.cn (M. Zhou), huanghw@tongji.edu.cn (H. Huang), 09zhang@tongji.edu.cn (D. Zhang), pzc2535314802@163.com (Z. Peng).

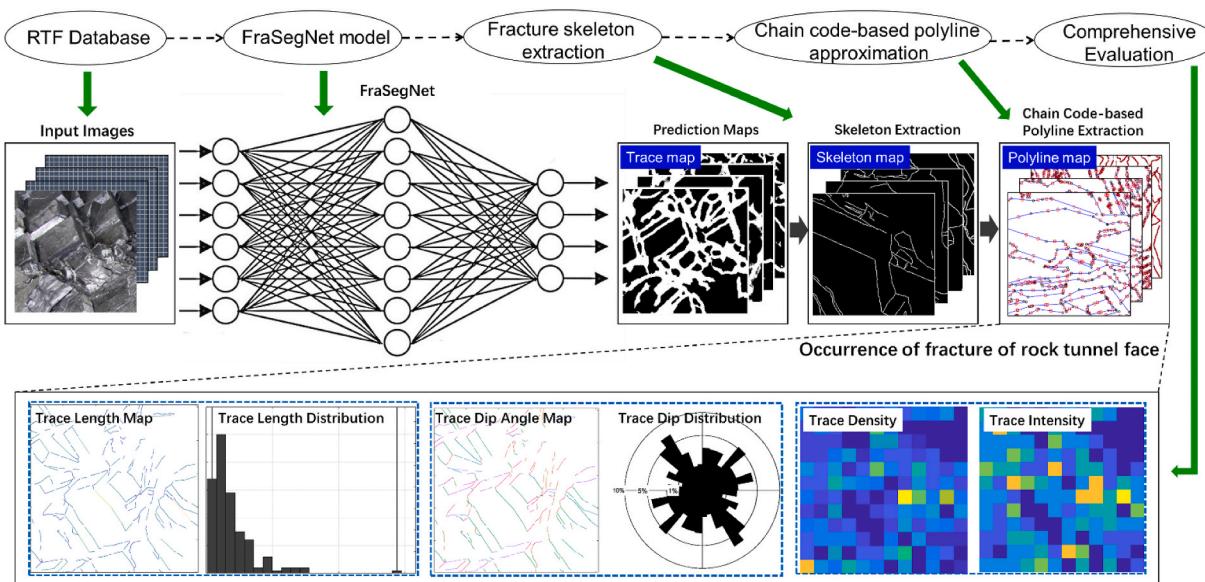


Fig. 1. Workflow of the proposed fracture trace extraction method.

backbones of those models (VGG,³⁶ ResNet,³⁷ Inception,³⁸ etc.), and the representative modules in the model (spatial pyramid pooling (SPP),³⁹ encode-decode,⁴⁰ conditional random field,^{32,41} atrous spatial pyramid pooling (ASPP),⁴⁰ etc.). Using these backbones and modules, researchers are committed to develop application driven CNN-based models solving feature segmentation tasks in tunnel inspection tasks, such as automated detection of lining cracks, surface spoiling, and water leakages.^{8,42–46} Although these models have shown promising performance,⁴⁷ the determination of the hyper-parameters of the feature extraction processes are highly dependent on the image resolution and specific feature characteristics.⁴⁸ It is thus necessary to re-structure the existing CNN-based models with well-defined application requirements, which considers the complexity of feature extraction and particularity of the fracture traces in the rock tunnel face images, and is able to extract the fracture trace map automatically.

Image-based fracture trace map evaluation depends heavily on locating lineaments of the extracted fracture skeleton features from the rock mass images.^{49–52} After obtaining the fracture trace map, the corresponding fracture skeleton map is required for comprehensive evaluation of the fracture traces. The fracture skeleton is defined as a collection of points with equal distance from the two edges of the fracture, that is, the central axis of each fracture region. There are two categories of methodology for locating the lineaments from images containing complex fracture skeleton features: the methods of transferring binary fracture image files to line-patterns, and the methods of identifying the key nodes of each fracture to form line-pattern files.⁴ In the first category, the Hough transform method⁵³ is commonly used to search co-linear patterns of pixels and produce fracture skeletons in the target image. However, it only performs well to identify single and thin linear features rather than in fracture traces consisting of multiple lines with non-uniform thicknesses.⁵⁴ In the second category, the key nodes in the fracture skeletons are often manually identified and labelled using an image editing tool (e.g. Adobe Illustrator). Then, the locations of the labelled key nodes are output to a text file and the key nodes can be connected to form the lineaments of the fracture skeletons. The fracture trace occurrences can be quantitatively evaluated based on the locations of the key nodes and the connected lineaments. Such a key node extraction process is a time-consuming objective, especially for complex multi-fracture cases.

Chain code-based contour extraction methods have been widely used for the efficient quantification of irregular features by systematically defining the main direction of the contour trend.^{55,56} However, the

output format of the chain code-based method does not provide the coordinates of key nodes. Thus, a polyline approximation algorithm is adopted in this study to automatically extract the coordinates of key nodes by an optimized Euclidean distance, and the processed coordinate information is output into a text file.^{57–62} Once the lineaments of the fracture skeletons are located from the output text file consisting of the key nodes coordinates, a fracture polyline map can be generated to evaluate the corresponding fracture traces. There are four typical fracture trace evaluation methods, including: line, rectangular window, circular window, and convex polygon window methods.^{52,62–64} Among these, the scan line method presents poor applicability for fractures with complex distribution and difficulty for a large-scale rock mass surface. The other three methods divide the target images into equal fictitious blocks with corresponding geometries to ensure a consistent envelope area. The circular window scanning method has the advantages of straightforward mathematical definition and wider adoption for calculating the fracture trace parameters.^{3,4}

In this work, a digital photography system is adopted to collect sample images of fractured rock mass exposures in rock tunnel faces. An image dataset, named the rock tunnel fracture (RTF) database, is then established containing 3,000 images with 1000×1000 pixels resolution. A novel method is proposed for automatically extracting 2D fracture trace maps from these rock tunnel face images and quantitatively evaluating the fracture trace maps. This proposed method is integrated using a CNN-based model named FraSegNet, a skeleton extraction algorithm, and a chain code-based polyline approximation algorithm. A comprehensive analysis is performed using various different feature extraction methods to show the superior performance of the proposed FraSegNet model. The polyline and the corresponding control node coordinates for each fracture skeleton of the fracture trace map are obtained by applying the skeleton extraction algorithm and a chain code-based polyline approximation algorithm. The circular scanning window evaluation method is used to analyse the extracted 2D fracture polyline map, and the evaluation accuracy of the proposed method is compared against the labelled dataset. A case study is then conducted on three full-scale tunnel face images obtained from the Mengzi-Pingbian Highway project site in Yunnan, China.

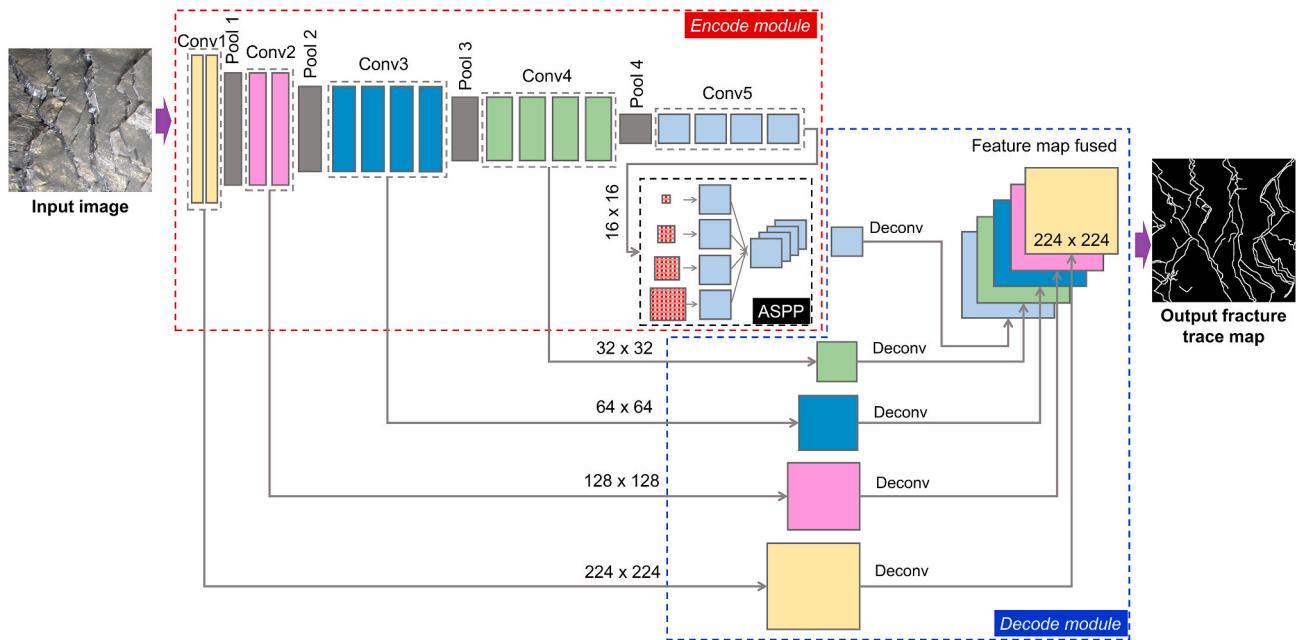


Fig. 2. Schematic diagram of the proposed FraSegNet model.

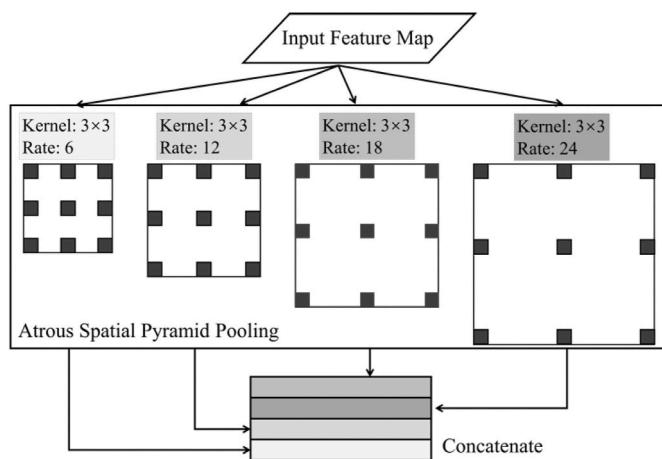


Fig. 3. Schematic of the adopted ASPP module.⁴⁰

2. Proposed method

2.1. Workflow of the proposed method

The proposed method for 2D fracture trace map extraction and evaluation is illustrated in Fig. 1, where the workflow comprises five components: (a) the RTF database, which contains the training and testing images as well as the ground truth fracture skeleton maps (see Section 3); (b) the FraSegNet model for fracture trace map extraction (see Section 2.2); (c) fracture skeleton map extraction (see Section 2.3.1); (d) chain code-based polyline approximation, which generates the fracture polyline map and obtains the key nodes' coordinates of the fracture traces (see Section 2.3.2); and (e) evaluation of the extracted fracture traces, including the following key characteristics of the fracture traces: the statistical information of the trace length, the trace dip angle, the trace density, and the trace intensity (see Section 2.4).

2.2. The FraSegNet model

2.2.1. Model structure

An ideal CNN-based model demands reliable performance at feature extraction and a unique network structure that can obtain quantified high-level features (i.e., pixel value, shape, texture, and position). Fig. 2 shows the proposed FraSegNet model, which consists of an input layer, a CNN-based encode module, a decode module, and an output layer. The encode module is modified from the convolution neural network VGG19³⁶ including five convolution layers (conv1, conv2, conv3, conv4 and conv5) and four pooling layers (pool1, pool2, pool3, pool4). In total, the modified VGG19 model consists of twenty hidden layers, including sixteen convoluted layers and four pooling layers. The structure is relatively simple since the whole backbone uses the same size as the 3×3 convolution core. Among them, the convolution layers are used for the purposes of feature extraction, and a convolution kernel is adopted in each layer to conduct a convolution operation, extract deeper features, and to pass the output features to the subsequent convolution layer. The weight and bias parameters of each convolution layer are used by the backpropagation algorithm to learn the features. Each convolutional layer is followed by a batch normalization to reduce the covariate shift and a rectified linear unit (ReLU) function to accelerate the convergence. A pooling layer is adopted to avoid overfitting by reducing the size of the feature map and the model parameters.

After four consecutive stages of convolutional and pooling layers, the feature map output from the fifth convolution layer is input into the atrous spatial pyramid pooling (ASPP) module (see Fig. 3).⁴⁰ The ASPP module is adopted to sample the feature map (output from the fifth convolutional layer) in parallel with the atrous convolutions at four different sampling rates. As a result, the sub-region representations at multiple scales can be fully captured. The side-output feature maps from conv1 to conv4 with different feature scales are also meaningful for the final feature prediction. More structure features persist in the shallower convolution layers, whereas the deeper convolution layers extract more abstract semantic features. It is vital to pass the five obtained feature maps from the five convolution layers (conv1, conv2, conv3, conv4, and conv5 with the ASPP) to the decode module.

The decoder module applies multi-scale up-sampling operations to ensure that the feature map sizes output from each convolution layer are consistent with the initial input image size. Fig. 4 demonstrates the five

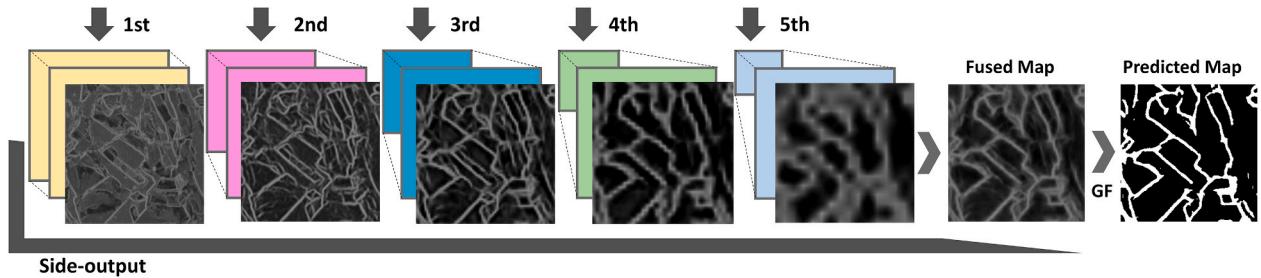


Fig. 4. An illustration of side-output predictions from the five convolution layers, a fused feature map predicted from the five outputs, and the final predicted fracture trace map processed by a guided filter (GF) operation.

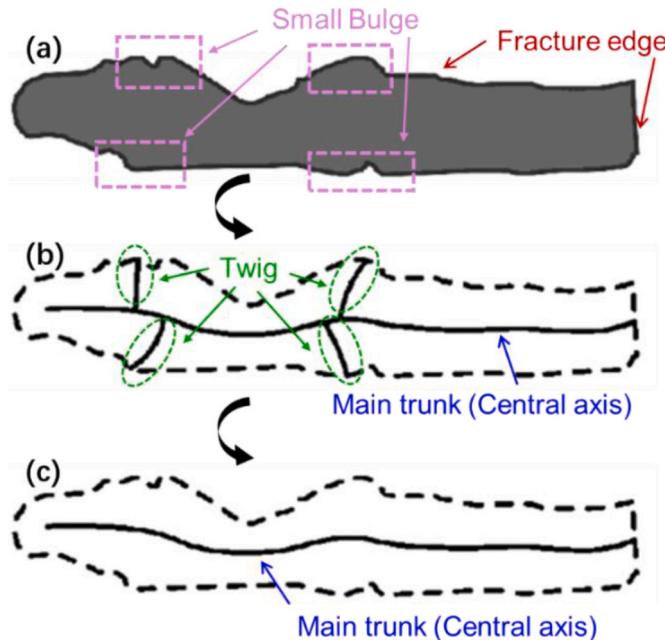


Fig. 5. The main process of fracture skeleton map extraction and optimization: (a) identify the fracture edge and small bulges, (b) extract the central axis of the fracture traces and the twigs, and (c) apply a clipping algorithm to remove the twigs.

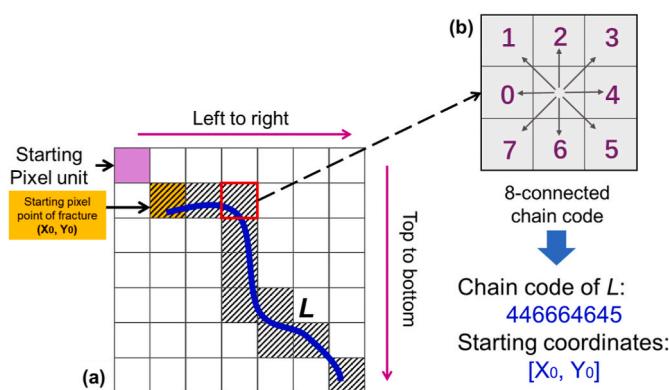


Fig. 6. Schematic diagram of the chain-code algorithm: (a) the ergodic coding process; (b) the relative positions of the pixel points.

feature maps output from the five convolution layers, a fused feature map predicted from the five outputs, and the final predicted fracture trace map processed by a guided filter (GF) operation. The adopted GF operation was proposed by,⁶⁵ which can ensure the edge-preserving

smoothing property and avoid the effect of noise brought by the output of the shallow convolution layers.⁶⁶

2.2.2. Model optimization

The FraSegNet model uses the stochastic gradient descent (SGD) algorithm as an optimizer, which randomly selects only one sample to update the model parameters within one training epoch. The SGD can thus efficiently minimize the computed loss function through iteratively updating the weights using a small batch of random selected training datasets. A cross-entropy loss function (Equation (1)) is defined to compute the difference between the manually labelled fracture trace map and the corresponding feature map predicted from the FraSegNet model:

$$L = - \sum_{i \in G_+} w_0 \log \Pr(P_i = 1) - \sum_{i \in G_-} w_1 \log \Pr(P_i = 0) \quad (1)$$

where w_0 and w_1 are the loss weights of the non-fracture and fracture categories, $\Pr(P_i)$ is the probability distribution of the predicted feature map, and G_+ and G_- denote the non-fracture pixels and the fracture pixels of an input image, respectively. Each convolution layer generates a prediction feature map and a corresponding side-output loss, termed $L_{\text{side-output}}^i$, where i is the layer number. The side-output layers are concatenated to a final fused map which produces the fused loss termed L_{fused} . The overall cross-entropy loss function is calculated as the summation of the loss values of the five convolution layers and the fused map:

$$L_{\text{overall}} = \sum L_{\text{side-output}}^{i=1 \text{ to } 5} + L_{\text{fused}} \quad (2)$$

The initial model hyper-parameters, namely batch size, learning rate, weight decay, momentum, and number of epochs, also need to be set before the training process is carried out.^{36,66} The batch size is the number of images fed into the target model for training. Learning rate refers to the magnitude of updating the network weight in the optimization algorithm. Weight decay is the coefficient of the regularization term in the loss function. Momentum is used to control the convergence rate of the model in the stochastic gradient descent (SGD) algorithm. An epoch refers to the process in which all the data is fed into the network to complete a forward calculation and back propagation. To optimize performance, the number of epochs should be adjusted to avoid overfitting or under fitting.

2.3. Evaluation of the fracture trace map

2.3.1. Extraction of the fracture skeleton map

Quantitative evaluation of the extracted fracture trace maps requires extracting the main trunk of each fracture and identifying the pixel-level nodes of the fracture traces. Fig. 5 illustrates the main process of fracture skeleton map extraction and optimization.^{67,68} The skeleton of a polygon is formed by a continuous shrinking process in which the edges of the polygon are moved inwards parallel to themselves at a constant speed. A considerable number of small bulges on the original fracture

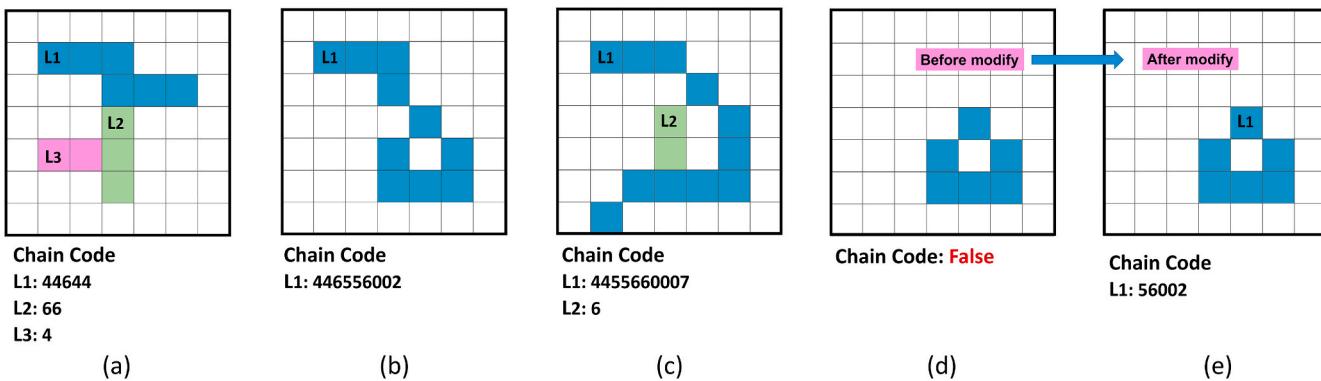


Fig. 7. Capability for coding special fracture distribution forms, including: (a) multiple bifurcations, (b) a circular structure at the end, (c) a circular structure in the middle, (d) a false single ring structure, and (e) a modified single ring structure.

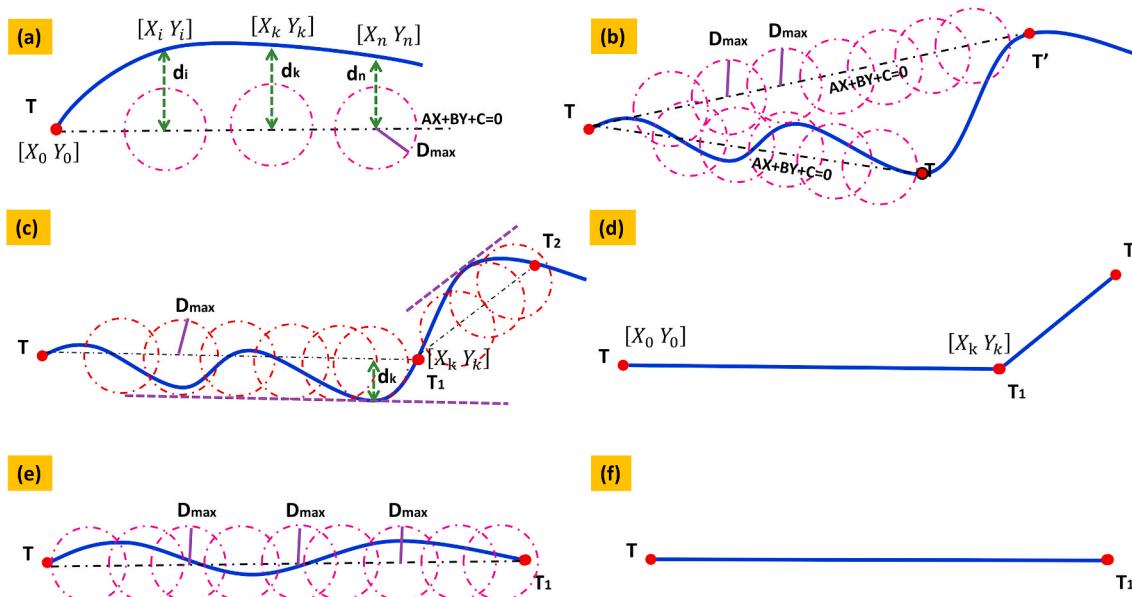


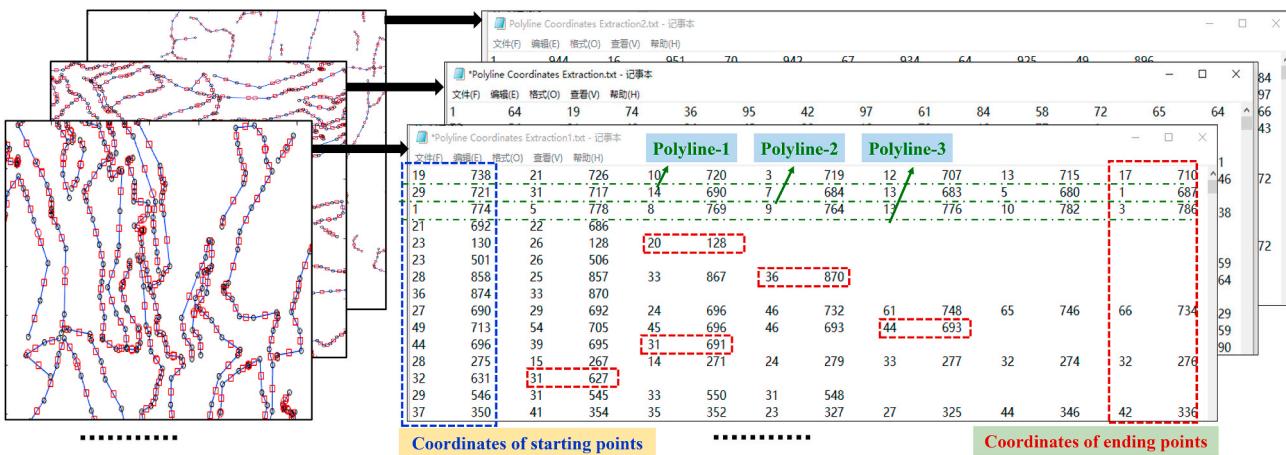
Fig. 8. The core processes of the polyline approximation extraction algorithm, including: (a) distance of the straight line between a point on the curve and two points of fracture, (b) calculation diagram of any points connected with the starting point in a straight line, (c) determination of the point satisfying the D_{\max} value, (d) the corresponding extracted polyline of (c), (e) all the Euclidean distances are less than the D_{\max} value, and (f) the corresponding extracted polyline of (e).

may form twigs in the skeletonization process resulting an inevitable operation error. A clipping algorithm is thus used to automatically remove the shorter branches. The extracted fracture skeleton map is an image with black and white colours only, where the white pixel points are the skeleton lines representing the fractures and the black pixel points represent the non-fracture regions.

The extracted fracture skeleton map provides the basis of the fracture trace geometry. However, it is challenging to quantify the complex fracture skeleton map as it contains a large number of pixel-level nodes and the curvature of each fracture varies. In this regard, a chain code-based polyline approximation algorithm is developed to automatically search for the decisive feature points in the fracture skeleton map. Each fracture skeleton is represented succinctly by a chain code to realize the polyline approximation operation. The chain code represents the specific fracture skeleton trajectory. As illustrated in Fig. 6 (a), the chain-code algorithm traverses all the pixel points of the fracture skeleton map from left to right and from top to bottom and assigns a chain code to each pixel point within a fracture skeleton. Through the ergodic process as shown in Fig. 6(a), the starting pixel-point of a fracture skeleton is

first identified and the coordinates of the starting pixel-point are stored as the heading of the chain code. Then, the next pixel-point in this fracture skeleton is searched and the corresponding chain-code is stored until the endpoint of the fracture skeleton is reached.

The next pixel-point position of the fracture skeleton is interpolated in the clockwise direction with respect to the surrounding 8 pixel points of the current position, and the corresponding numeric codes of the surrounding 8 pixel points are denoted as numbers 0 to 7 (Fig. 6(b)). If the pixel-point has been coded in a previous fracture skeleton, this pixel-point is skipped and moves in clockwise direction to the next neighbourhood pixel point until a pixel-point is reached which has not been coded and lies within a fracture skeleton. If the pixel-point has not been coded, the numeric code of the current pixel-point with respect to the previous pixel-point is interpolated, and then the next pixel-point in this fracture skeleton is found with respect to the current pixel-point, and so on, until the endpoint of this fracture skeleton is reached. A fracture endpoint is derived if the current pixel-point lies within the fracture skeleton and the following chain code number is 1. A new fracture starting point is searched when an endpoint of a fracture skeleton is



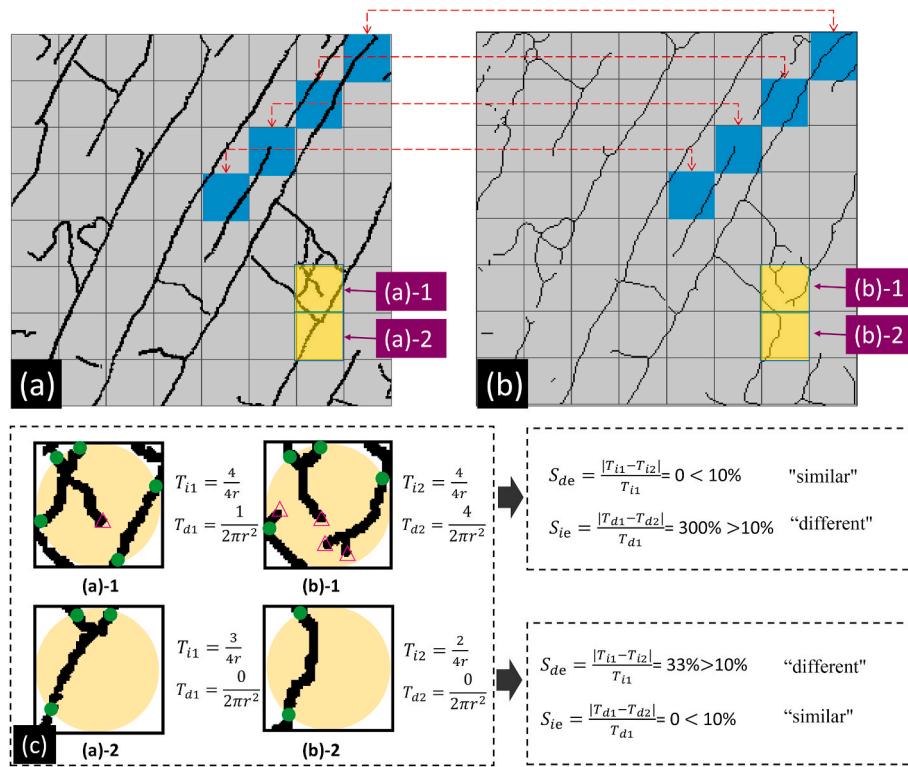


Fig. 11. Schematic diagram of trace intensity/density evaluation index calculation, including: (a) ground truth fracture polyline map, (b) predicted fracture polyline map, and (c) examples for computing the trace intensity and trace density indexes.

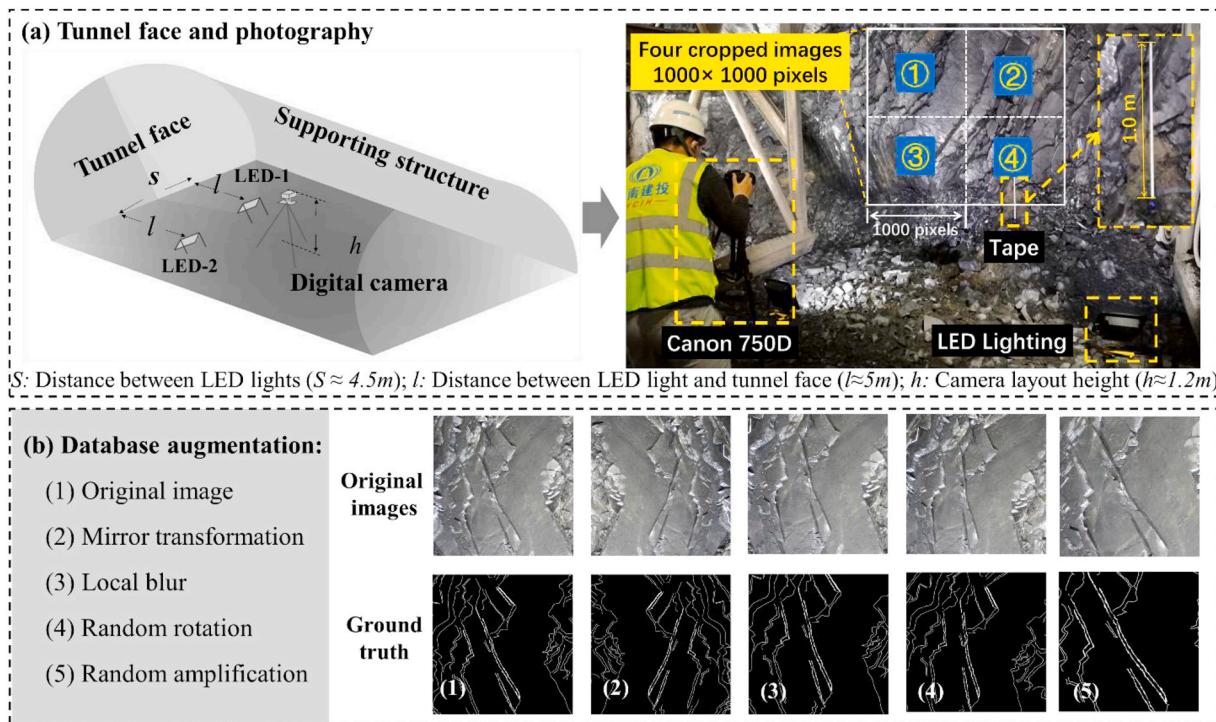


Fig. 12. Schematic diagram of database acquisition and establishment, including: (a) rock tunnel face and the photography system, and (b) image database augmentation.

defined as the lines connecting neighbouring key nodes of a fracture polyline. The trace dip angle is defined as the angle between a polyline and the horizontal direction to the right, which is computed as the statistical average of the dip angles of all the sub-polylines. As a result,

the statistical distribution of the trace lengths and the trace dips of all the polylines can be quantitatively evaluated.

As shown in Fig. 10, a two-dimensional grid composed of uniformly distributed circular scanning windows is generated to adapt to the

Table 1

The image dataset distribution for the fracture trace map extraction task.

Fracture category	Training	Testing	Total number
Fracture	2000	500	2500
Non-fracture	400	100	500
Total number	2400	600	3000

Table 2

Fracture feature extraction algorithms.

Deep Learning	FraSegNet-VGG16
	FraSegNet-VGG19
	FraSegNet-VGG16-ASPP
	FraSegNet-VGG19-ASPP
	DeepLabV3+
	FCN
Edge Detection	Canny
	Laplacian

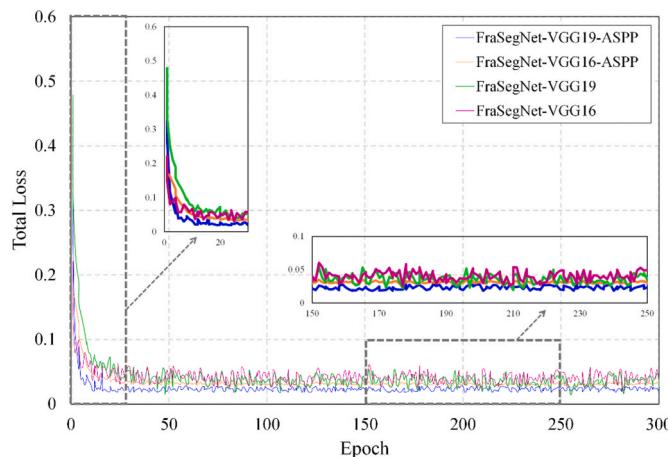


Fig. 13. The total losses for the four models with various network structure.

fracture polyline map area. The scanning circle diameter is defined as 0.99 times that of the grid spacing to avoid scanning circles overlapping. Then, the intersection number and ending pixel-points of the fracture polylines in the scanning circle are calculated by an ergodic statistical method, which is based on both the chain code and key node coordinate information.^{58,61,62}

By using the circular scanning window,⁵⁸ the trace intensity, T_i , is defined as the total trace length in a scanning window with a unit of m^{-1} (see Fig. 10):

$$T_i = \frac{n}{4r} \quad (6)$$

where n refers to the number of intersections between fracture polylines and the boundary of the scanning window, and r is the radius of a circular scan window. The trace density, T_d , is defined as the total number of fracture traces in a scanning window with a unit of m^{-2} (see Fig. 10):

$$T_d = \frac{m}{2\pi r^2} \quad (7)$$

where m refers to the number of fracture polyline ending pixel-points in a specific circular scanning window.

2.5. Performance indexes of the proposed method

The total loss is used to assess the training performance and the convergence speed of the FraSegNet model. The ground truth datasets

consist of the manually labelled fracture skeleton maps. The fracture feature extraction accuracy is not evaluated at the pixel level in this study since the fracture trace maps are skeletonized into fracture skeleton maps. It is difficult to compare the statistical results of the fracture traces between the ground truth (manually labelled fracture skeleton map) and the predicted fracture skeleton map. Therefore, the predicted and the ground truth fracture skeleton maps are both processed by the chain code-based polyline approximation algorithm to generate the fracture polyline maps. Three performance indexes based on the scanning window method are proposed to quantitatively compare the predicted and the ground truth fracture polyline maps, namely the length evaluation index (T_{le}), the dip angle evaluation indexes (A_{sp} and A_{ss}), and the intensity and density evaluation indexes (S_{ie} and S_{de}).

The maximum length of a continuous fracture trace has been taken as an indispensable evaluation index in some internationally recognized discrimination methods,^{69–72} such as the rock mass rating (RMR). Therefore, the maximum trace length in a fracture polyline map is considered as the length evaluation index (T_{le}). The trace dip angle reflects the sliding tendency between structural planes, which affects the stability of the structural planes to a great extent.^{13–15,52} The primary dip angle, A_{sp} , and the secondary dip angle, A_{ss} , represent the most active fracture interactions between structural planes. Therefore, grasping the statistics of A_{sp} and A_{ss} is of great significance to evaluate the overall distribution of the fracture traces. As illustrated in Fig. 11, the trace intensity and the trace density in each scanning window are compared between the predicted and the ground truth fracture polyline maps. A specific window is considered to be “similar” whenever the difference in the trace intensity or the trace density is less than 10%. The proportion of similar windows is calculated by comparing the information from all the grids in a fracture polyline map. The counted similarity proportions, S_{ie} and S_{de} , are the trace intensity and the trace density evaluation indexes, respectively.

It should be noted that there may be uncertainties for the image-based validation since the images obtained from the tunnel site may not represent the real geometric information. It would be ideal to have on-site compass measurement data to validate our proposed method. However, it was difficult to carry out the measurement due to the harsh on-site environment. Fortunately, the ground truth images with labelled fractures contain a wealth of domain knowledge from the rock tunnel experts, which to some extent represents the real conditions of the tunnel face and can be used for performance validation of this study.

3. Image database description

As illustrated in Fig. 12, the RTF database was established via a digital photography system, which takes rock tunnel face images from the Mengzi-Pingbian Highway (MPH) project. The photography system included a Canon 750D camera, measuring tools, a tapeline, and two 1000W LED lighting kits (Fig. 12(a)). During the image acquisition process, it should be ensured that the LED illumination is parallel and uniform, the distance between two lights is an approximate value of 4.5 m, and the distance between camera and tunnel face is controlled with an approximate value of 5.0 m. To quantify the geometric information of fracture traces, the pixel size calibration is estimated from the size of a pixel and the statistical number of pixels.⁴⁶ For this purpose, the size of a pixel is derived by placing a tapeline on the tunnel face as a benchmark reference (Fig. 12(a)), and the statistical number of pixels is obtained by the deep learning-based segmentation. More than 550 raw images with a maximum resolution of 6000 × 4000 pixels were collected from various tunnel sites in MPH. These raw images were then selected and cropped into 2000 smaller images (1000 × 1000 pixels). The corresponding fracture skeleton maps of the 2000 smaller images were manually labelled via Photoshop software. It is worth noting that the fracture skeletons are assumed to be abstract lines without width, and to exist where the rock splits with no obvious relative displacement on both sides. Therefore, the width of the labelled fracture skeleton is set as one

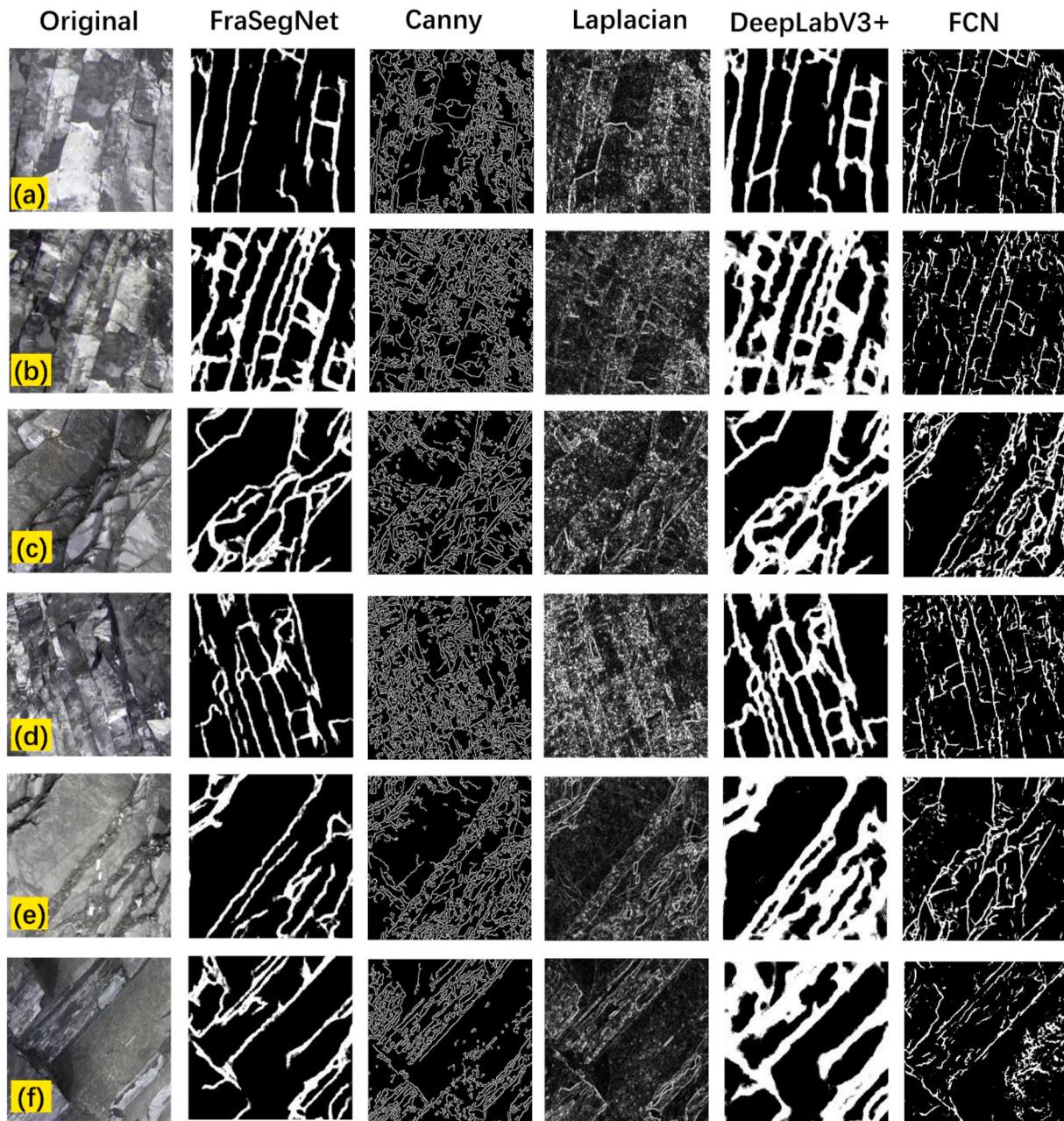


Fig. 14. Examples of extracted fracture trace maps using various methods, including deep learning and traditional edge detection methods.

pixel. Field engineers are more concerned with fractures of natural origin rather than cracks caused by blasting.^{61,62} The labelled images in this study were checked by experienced field engineers and tunnel experts to ensure their consistency and quality. Therefore, these blasting-caused fractures were not labelled in the “ground truth” dataset. Although the labelling process is somewhat subjective, interactive operation with the labelling software can reduce marking errors until they meet the engineer’s standard. Thus, the manual labelling method can be reasonably applied in this study.

To enrich the RTF database, augmentation techniques (Fig. 12(b)), such as mirror transformation, local blur, random rotation, and random amplification were performed.^{46,73} Fig. 12(b) shows how four new images and their corresponding labelled images are generated synchronously from a typical original image using the aforementioned augmentation techniques. As summarized in Table 1, a total number of 3000 images were produced containing 2500 and 500 fracture and non-fracture samples, respectively. For each category, 80% of the

samples were used for training and the remaining 20% for testing. To determine the total manual labelling time, each original image with fracture occurrences was augmented four times to produce a dataset containing 2000 images. Therefore, 400 images with fracture occurrences were labelled in this study to setup the dataset. Labelling each image takes about 3 min using the Photoshop software. It thus takes about 1200 min to complete the labelling process, and image augmentation takes only a few minutes using the compiled python codes.

4. Experiments and results

The established RTF database was trained and tested on an integrated workstation configured with a 32 GB RAM, NVIDIA GeForce GTX-1080 Ti, and Intel Core i7-8700 GPU processor. The Python programming language was compiled in Windows 10. The training and testing processes were performed on the corresponding portion of the RTF database as suggested in Table 1.

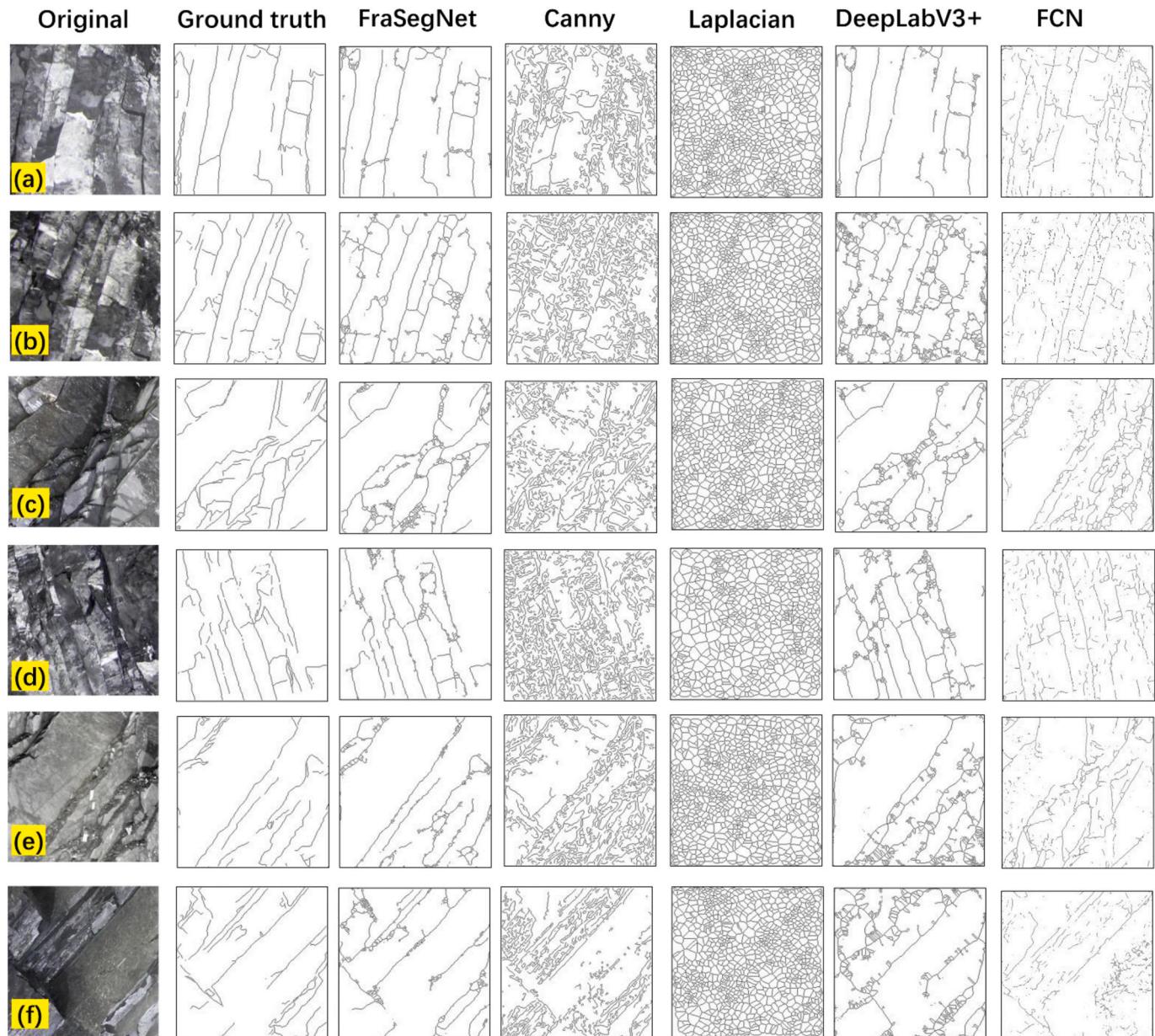


Fig. 15. Examples of fracture skeleton maps using various methods, including ground truth labelling, deep learning, and traditional edge detection methods.

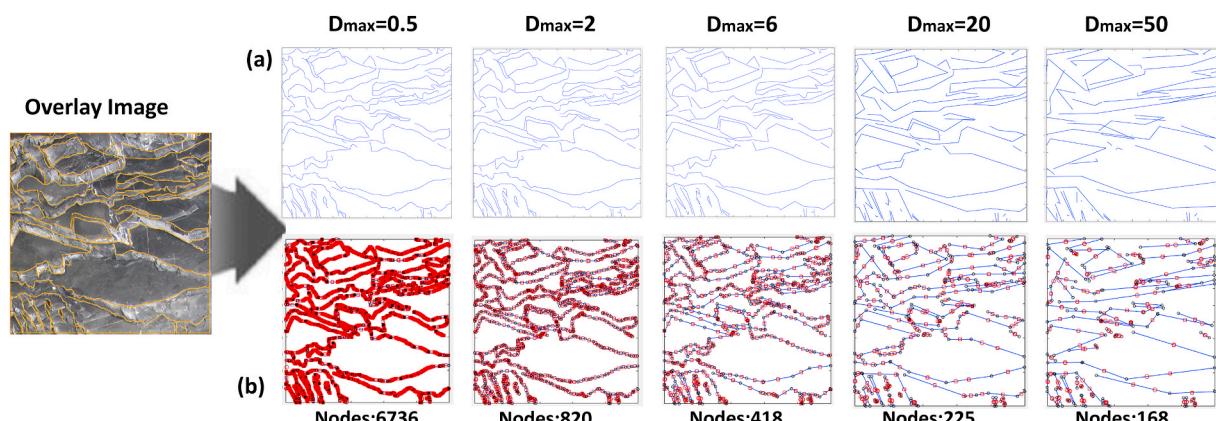


Fig. 16. Results of fracture polyline maps with D_{\max} values from 0.5 to 50 pixels, including: (a) fracture polyline map, and (b) extracted nodes-map for various D_{\max} values.

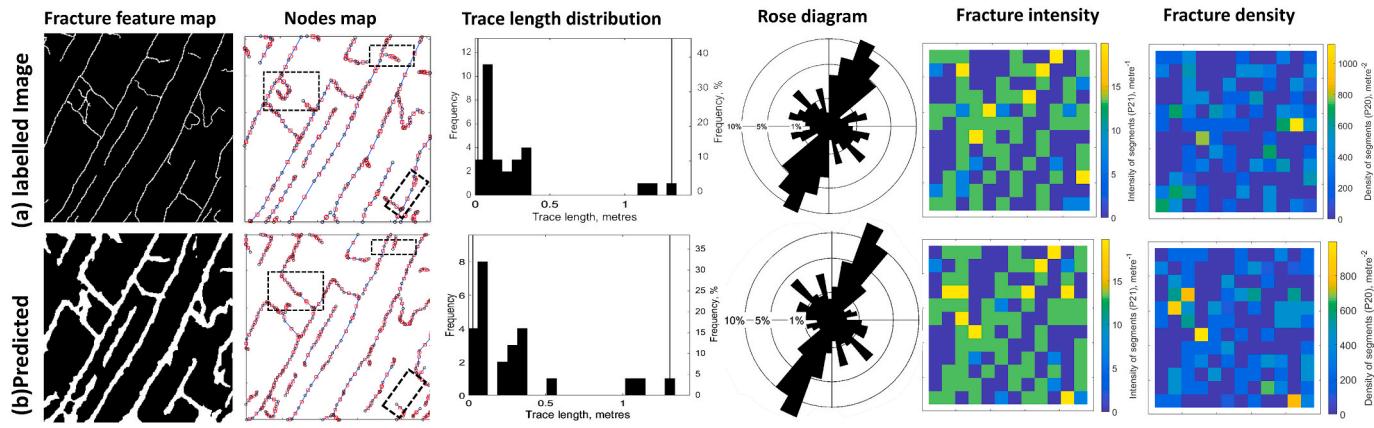


Fig. 17. Comparison of evaluation results of the ground truth and the prediction for a typical testing sample.

Table 3

Statistical results between the ground truth and the prediction of a typical testing sample.

Data type	Key Nodes numbers	Trace length distribution		Trace dip angle (°)	
		Mean (m)	Variance	Primary	Secondary
Ground truth	195	0.268	0.0915	67.3	142.6
Predicted	232	0.316	0.0998	67.1	141.5

4.1. Fracture trace map extraction and skeletonization

Four various network structures were trained and tested to obtain an optimized FraSegNet model; the adopted backbone and modules are summarized in Table 2: (1) FraSegNet-VGG16 applied a VGG16-based backbone and without an ASPP module behind the Conv5; (2) FraSegNet-VGG19 applied a VGG19-based backbone and without an ASPP module behind the conv5; (3) FraSegNet-VGG16-ASPP applied a VGG16-based backbone and with an ASPP module behind the conv5; (4) FraSegNet-VGG19-ASPP applied a VGG19-based backbone and with an ASPP module behind the conv5.

As defined in Section 2.2.2, the hyper-parameter values are set to be the same for the four FraSegNet-based models in order to conduct reasonable comparisons. The four models were all trained at an initial learning rate of 0.0001, momentum of 0.9, weight decay of 0.0001, a batch size of 1, and a loss weight of 1.0 for each side-output and the final fused layer. The training epoch was set at 300 to avoid non-convergence

caused by the insufficient training step. The learning rate was changed to the original 0.1 after every 30 epochs. All of the training processes were performed at the same workstation configured with the aforementioned parameters.

The total loss curves of all four models against the training epochs were recorded during the training process. As illustrated in Fig. 13, the total losses sharply decreased at the beginning of the 8th epoch, and then converged steadily from 25 epochs and beyond. Of the four models, the FraSegNet-VGG19-ASPP model presented the least total loss in the training process. It is thus concluded that applying a deeper backbone and an ASPP module can both handle a more ideal training performance. Therefore, the FraSegNet-VGG19-ASPP model structure is considered to be the optimized FraSegNet model for extracting fracture trace map.

At present, the traditional feature-based extraction method and deep learning-based method are two main commonly applied methods in engineering. Thus, representative algorithms of each corresponding method were selected to conduct comprehensive comparisons. Two traditional image edge detection algorithms (Canny and Laplacian) and two deep convolution neural network (DCNNs) methods (DeepLabV3+ and FCN) are illustrated in Table 2 and were implemented in the training process for further performance comparison against the proposed FraSegNet model. The basic training parameters and datasets, as well as the operating equipment for the DCNNs-based approaches, were the same as those used in the FraSegNet model to ensure consistency. Fig. 14 shows the feature extraction results of the fracture trace maps using various methods on six different testing samples. By qualitative comparison, the FraSegNet model performed more effectively than the FCN and traditional edge detection algorithms (Canny and Laplacian) in terms of boundary recognition and noise avoidance at the pixel level. In other

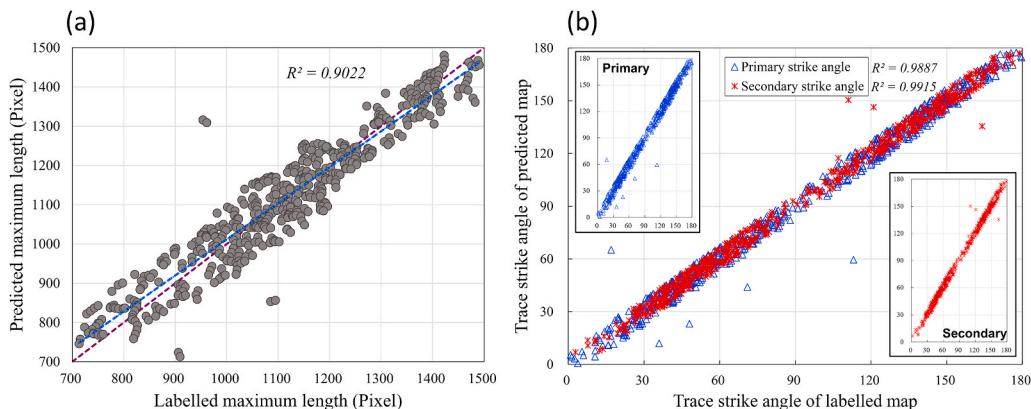


Fig. 18. Comparing the evaluation indexes of the predicted and labelled fracture skeleton maps, including: (a) the maximum trace length, and (b) primary and secondary dip angle.

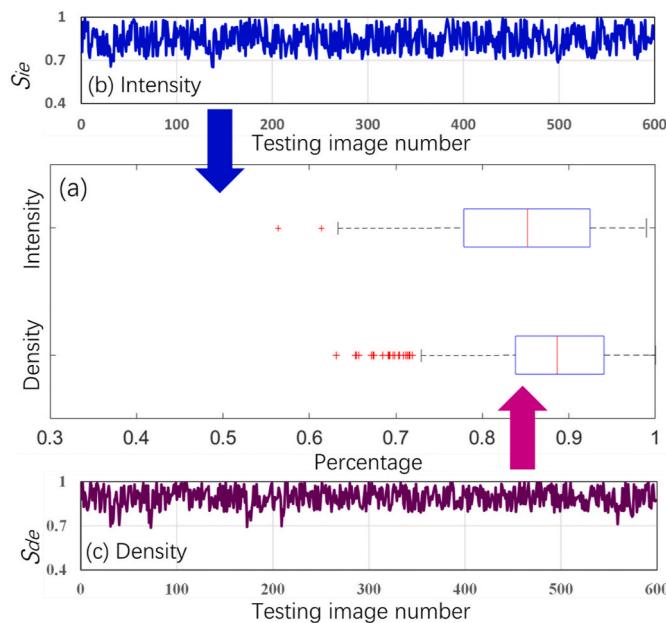


Fig. 19. Results of the intensity and density similarity evaluation indexes for the testing dataset.

words, although the traditional method can appear to accurately segment the fracture according to the image texture by adjusting the threshold, the FraSegNet model can more deeply learn the meaning of the real fracture conveyed by the informed domain knowledge. Meanwhile, as a deep learning framework, the FCN model cannot effectively avoid the interference caused by fragmentation. The DeepLabV3+ model showed similar feature extraction performance to the FraSegNet model but over-estimated the width of the fracture traces.

Skeletonization is an important step after extracting the fracture trace map. Fig. 15 exhibits the visualization results of the fracture skeleton maps based on the ground truth labelling and the extracted fracture trace maps (shown in Fig. 14) using various methods. It was found that the performance of the deep learning-based algorithms was significantly better than that of the image edge detection algorithms. By visual comparison with the ground truth, the DeepLabV3+ model and the FCN model presented a large number of noise lines and missing fracture skeletons. Overall, the FraSegNet model had the best performance with few noise lines at the intersections of the fracture skeletons.

It should be noted that a diversified database is key to ensuring that the deep learning-based framework can be used in different tunnels. That is, during a given CNN-based feature extraction task, a dataset only collected from a specific region or tunnel may limit its applicability for fracture characterization at other sites. Increasing the diversity and amount of image samples is important to improve the fracture extraction performance at different tunnel sites. The database established in this study can be undoubtedly applied for the tunnel sites in the MPH region. However, when encountering samples that are very different from the existing datasets, the performance of fracture segmentation may not be as good. In that case, users can further enrich the current database for wider use of automatic fracture extraction.

4.2. Evaluation of the testing samples

Based on the fracture skeleton map and the chain-code based polyline approximation procedure, the key nodes of the fracture traces were obtained in a text file. The D_{max} values affected the number of extracted key nodes and the similarity between the approximated fracture polyline map and the ground truth. Fig. 16 shows the fracture polyline maps of a testing sample and the corresponding key nodes amount with D_{max} values from 0.5 to 50 pixels. The results suggested that a smaller D_{max}

value (less than 2 pixels) may over emphasise the detailed radius of the fracture traces and lead to errors in the evaluation of the fracture traces, whereas a larger D_{max} value (greater than 20 pixels) may cause distorted fitting since most fracture traces are approximated as a straight line without capturing the curvature of the fractures. As a result, the D_{max} value is set to 6 pixels in this work to produce a reliable fracture polyline map with a reasonable number of key nodes.

By applying a D_{max} value of 6 pixels, the fracture polyline maps of the 600 testing samples were thus generated and the extracted key nodes of each fracture polyline map were used to evaluate the fracture traces in the testing images. Fig. 17 and Table 3 demonstrate the visualization and corresponding statistical results of a typical testing sample and its corresponding ground truth benchmark, respectively. The fracture polyline maps had 195 nodes from the labelled skeleton map, which was 37 nodes fewer than the 232 nodes of the predicted fracture skeleton map. It can be seen that the two fracture polyline maps are similar except for the regions in the black dotted boxes marked in Fig. 17. For the trace length distribution, the two maps showed similar variance for a log-normal distribution, but had some discrepancies in the mean length value. The rose diagram, symmetric at the origin, reflects the dip angles of the fracture traces. The main dip angle of the fracture trace lay from northeast to southwest, and that of the secondary dip lay from northwest to southeast. As shown in Table 3, the dip angle values of the primary and the secondary directions were almost unanimous for the ground truth and the predicted maps. The intensity and density of the fracture polyline maps was evaluated by the scanning window method. The radius of the scanning window was set as 25 pixels, which divided each fracture polyline map into 400 (20 × 20) cells with a grid size of 50 × 50 pixels. The proposed performance indexes were used for quantitative comparison of the intensity and density results of all 600 testing samples.

4.3. Quantifying the performance of the proposed method

The performance indexes of fracture length, dip angle, intensity, and density proposed in Section 2.5 were used to compare the statistical results of the 600 testing samples and to quantify the performance of the proposed method. The T_{le} (maximum trace length) values of the 600 predicted fracture polyline maps were plotted against the ground truth values obtained from the maps generated from the manually labelled fracture skeleton maps (Fig. 18 (a)). A reference line with gradient of 1 is included on the plot, and a linear regression line is also plotted with a gradient of 0.92 and R^2 of 0.9. The predicted T_{le} depends on the continuity of the connection of the key nodes, that is, the existence of breakpoints will stop the statistic of the maximum length. It is feasible to tolerate a relatively small statistical error for the maximum trace length. In general, the prediction of the maximum trace length can effectively reflect the ground truth. Fig. 18 (b) presents the statistical results of the primary and secondary dip angles for the 600 testing samples. An ideal calculation accuracy was maintained between the predicted and labelled fracture dip angles. The statistical results for the angle were better than that of the fracture length. The potential reason for this is that the angle represents a description of the fracture morphology, while the length reflects the characteristics of fracture continuity. Overall, the fracture continuity is more difficult to quantify than the fracture morphology.

The S_{ie} and S_{de} indexes, the trace intensity and trace density similarity ratios, are calculated by comparing the relative intensity/density in a scanning window. Fig. 19(b) and (c) exhibit the distributions of S_{ie} and S_{de} indexes for the 600 testing samples and showed great fluctuation in the various samples. The box-plot Fig. 19(a) shows the overview statistical result, from which conclusions can be drawn that the average values of S_{ie} and S_{de} were 85.7% and 88.65%, respectively. Despite some data volatility, most of the fracture density and intensity can be accurately described by the proposed approach.

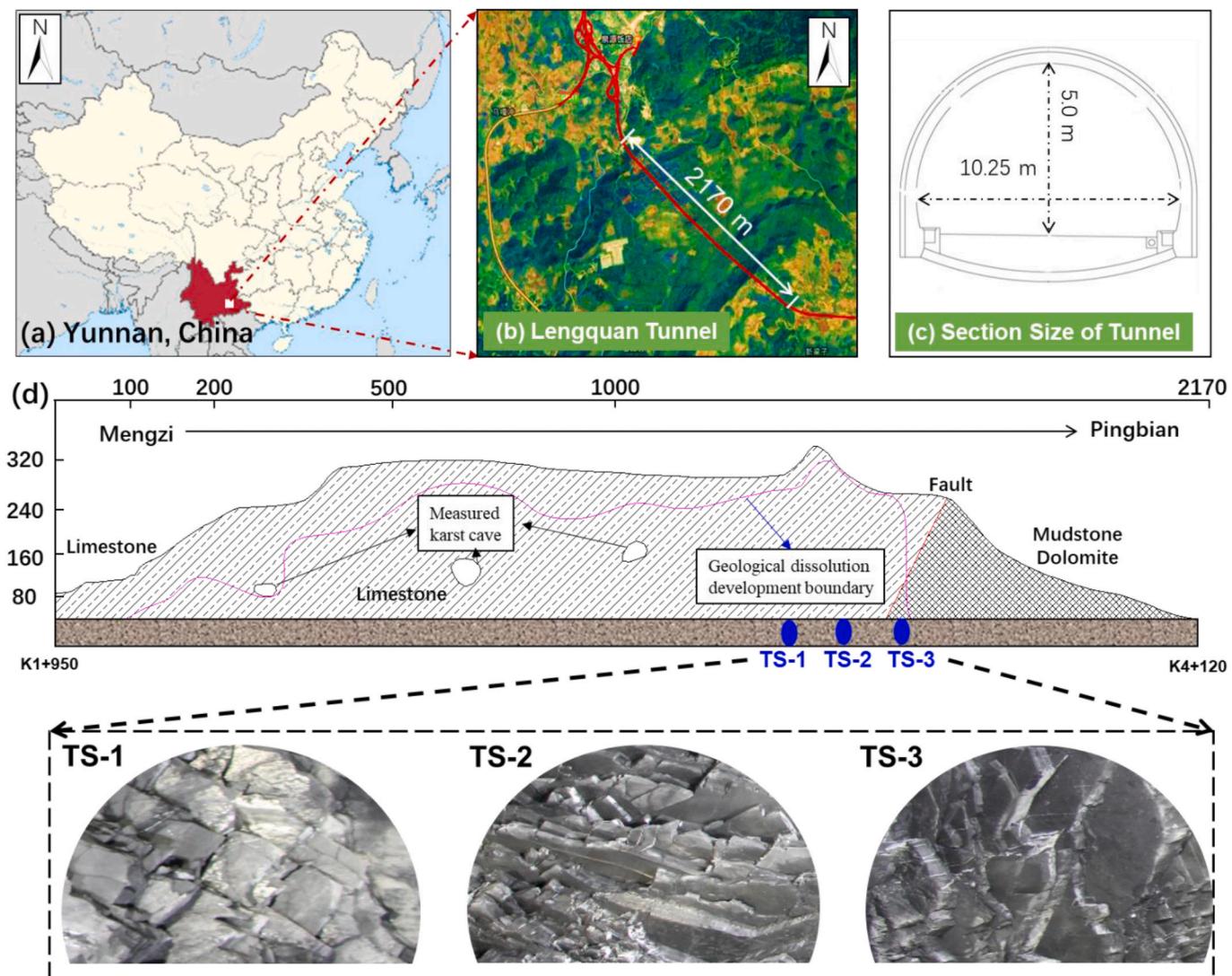


Fig. 20. The main information of the case study tunnel site, including: (a) the location, (b) satellite map, (c) tunnel section size, and (d) tunnel section geology and three selected tunnel workface sections.

5. Case study on full scale rock tunnel face images

The Lengquan tunnel is one of the most fractured rock tunnels in the Mengzi-Pingbian Highway Project, Yunnan, China (Fig. 20). It is designed as double tunnels with a height of 5.0 m, a width of 10.25 m, and a maximum buried depth of 312 m. The right line of the tunnel has an approximate total length of 2170 m, with the starting and ending stake numbers being K1+950 and K4+120, respectively. Through the preliminary investigation, it was found that the karst geological characteristics of the tunnel are complex and the fissures are extremely developed. The main rock categories around the Lengquan tunnel include moderately weathered dolomite, limestone, and intercalated weak mudstone. In the case study, the images of three tunnel workface sections (i.e., TS-1, TS-2, and TS-3 in Fig. 20(d)) were selected for the fracture trace map extraction and evaluation.

To ensure the rationality of the statistics of the whole face, especially for the continuity of trace length, it is necessary to eliminate length interruptions caused by manual image cutting. In the process of deep learning-based fracture extraction, the sub-images (1000×1000 portion of a full-size image) are initially cropped in order and used as input files for the deep learning model. The extracted sub-images are then reassembled with their corresponding orders to create the full-size image with extracted fractures. The fracture trace lengths are thus

computed based on the reassembled full-size image, so that the trace length of a truncated feature can be counted by connecting the sub-images.^{1,74}

The predicted fracture trace maps (Fig. 21(a)) and the fracture skeleton map (Fig. 21(b)) of the three tunnel faces were extracted using the proposed method configured with the optimized operating parameters, software environments, hardware settings, and operating time listed in Table 4. Data labelling and training only needed to be performed once to get the corresponding weights and biases for the proposed deep learning algorithm. The execution time was 0.44 s for image testing and 7.2 s for statistical evaluation. By qualitative inspection, the main fracture traces in the tunnel faces were well characterized (Fig. 21(c)). However, there were short fractures segmented from the extremely fractured regions and circular fracture traces at the intersection regions. This extracted “noise” can be avoided tremendously by setting a proper D_{\max} value (6 pixels was used in this case study) in the process of chain code-based polyline approximation. Based on the extracted key node coordinates and the generated fracture polyline maps of the three tunnel faces, the basic information (length and angle) of each fracture trace can be quickly obtained. The distributions of the trace length and the dip angle of the three tunnel face sections are shown in Fig. 21(d) and (e). It can be concluded that all the trace lengths are less than 2 m and most fracture traces have a length less than 1.2 m. The primary dip angles of

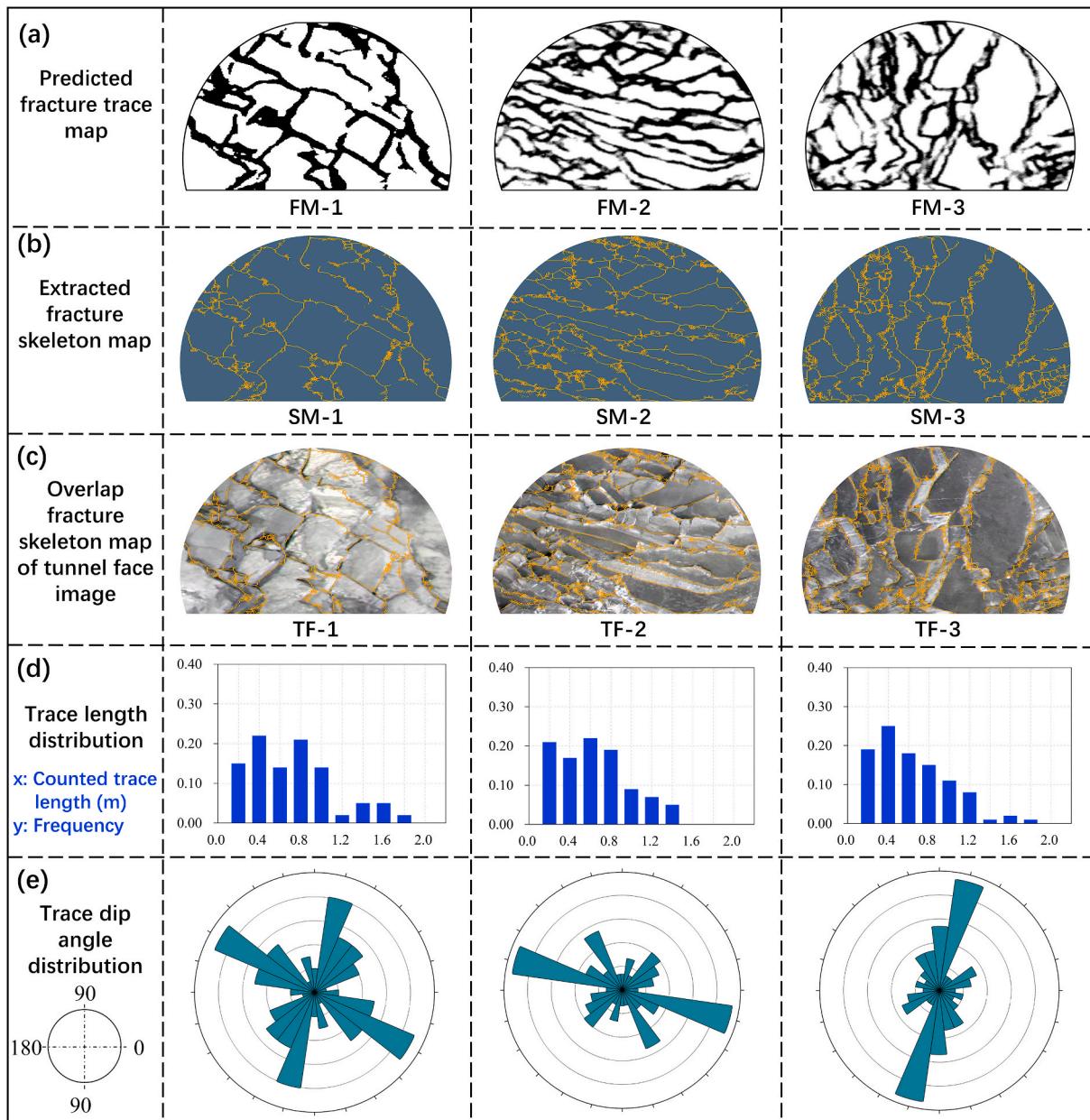


Fig. 21. Case study of the in-field rock tunnel face including: (a) predicted fracture trace map, (b) extracted fracture skeleton map, (c) overlap fracture skeleton map of tunnel face image, (d) trace length distribution, and (e) trace dip angle distribution.

the three sections are significantly different, which further demonstrates the complexity and rapid change of discontinuity in the tunnel faces. Overall, the proposed method can accurately extract the fracture trace maps and efficiently evaluate complex geological information in rock tunnel faces.

6. Conclusion

A novel image-based approach is proposed for automated fracture trace map extraction and evaluation from rock tunnel face images. It is integrated by a CNN-based model, named FraSegNet, a skeleton extraction algorithm, and a chain code-based polyline approximation algorithm. A total of 3,000 images were created to establish the RTF database and randomly selected for training and testing.

Comparisons with various backbones (VGG16 and VGG19) and modules (with and without ASPP) indicate that applying a deeper backbone and an ASPP can show a more ideal performance for the

FraSegNet model. Meanwhile, comparisons with DeepLabV3+, FCN, and the traditional image edge extraction algorithms (Canny and Laplacian) illustrate that the optimized FraSegNet-VGG19-ASPP method outperformed FCN and all the traditional edge extraction algorithms in terms of the prediction metrics, and presents an equivalent performance to DeepLabV3+. Throughout all the mentioned methods, FraSegNet-VGG19-ASPP shows the best performance in fracture segmentation and skeletonization.

The 600 images of the RTF database were selected, tested and processed to obtain the trace occurrence information. By applying the skeletonize algorithm and the chain-code based polyline approximation approach, the statistical trace information was compared for each group of predicted and labelled maps. Then, three quantitative evaluation indexes of fracture length, dip angle, intensity, and density of the fracture traces were first proposed and defined to compressively assess the performance of the proposed method. Qualitative and quantitative analysis shows that the proposed method can extract the trace occurrence

Table 4

The main configurations and parameters used in the proposed method for the case study.

Hardware configuration	
CPU	Intel(R) Core(TM) i7-8700 @ 3.70 GHz
GPU	NVIDIA GeForce GTX-1080 Ti
Memory	2T
RAM	32G
Software configuration	
Compiler language	Python, Matlab
IDE	PyCharm, Matlab
Other tools	Office 2019, Adobe illustrator
Operating parameters	
Model	FraSegNet-VGG19-ASPP
Initial Learning rate	0.0001
Momentum	0.9
Batch size	1
Epoch	200
D _{max}	6 Pixels
Image Size	1000 × 1000 Pixels
Operating time	
Data Labelling	1200 min
Training	8h 23min
Testing (s/image)	0.44 s
Statistics (s/image)	7.2 s

effectively and accurately, and is highly consistent with the labelled ground truth.

However, segmentation error in the process of deep learning recognition and the potential statistical error caused by the chain code-based algorithm may have an unavoidable impact on the final results. Meanwhile, the discontinuous nodes and interference short lines caused by the skeleton extraction algorithm still need to be optimized in future work. Moreover, it should be noted that the performance of deep learning-based fracture characterization is heavily influenced by the diversity and size of the established database. A diversified database is key to ensuring that the deep learning-based framework can be used in different tunnels. Further expanding the database in future work can improve the comprehensive applications of the proposed method in various projects. Although the whole processes of trace characterization are complex due to poor program integration, users can get satisfactory results when the reliability of sample quantity and quality is guaranteed. Finally, although it is not possible to achieve the ultimate rock mass classification using only the extracted fracture trace information, this study provides a step towards automated rock mass classification since the fracture trace information is one of the key components in all rock classification systems.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: The research study presented in this paper was supported by the Natural Science Foundation Committee Program of China (Grant No. 51778474), Science and Technology Project of Yunnan Provincial Transportation Department (No. 25 of 2018), and key innovation team program of Innovation Talents Promotion Program by the Ministry of Science and Technology (MOST) of China (No. 2016RA4059).

Acknowledgments

The research study presented in this paper was supported by the Natural Science Foundation Committee Program of China (Grant No. 51778474), Science and Technology Project of Yunnan Provincial Transportation Department (No. 25 of 2018), and key innovation team program of Innovation Talents Promotion Program by the Ministry of Science and Technology (MOST) of China (No. 2016RA4059).

References

- Chen J, Yang T, Zhang D, Huang H, Tian Y. Deep learning based classification of rock structure of tunnel face. *Geosci. Front.* 2021;12(1):395–404.
- García-Luna R, Senent S, Jurado-Piña R, Jimenez R. Structure from Motion photogrammetry to characterize underground rock masses: experiences from two real tunnels. *Tunn Undergr Space Technol.* 2019;83:262–273.
- Fadakar Alghalandis Y. ADFNE: open source software for discrete fracture network engineering, two and three dimensional applications. *Comput Geosci.* 2017;102:1–11.
- Healy D, Rizzo RE, Cornwell DG, et al. FracPaQ: a MATLAB™ toolbox for the quantification of fracture patterns. *J Struct Geol.* 2017;95:1–16.
- Zeeb C, Gomezrivas E, Bons PD, Virgo S, Blum P. Fracture network evaluation program (FraNEP): a software for analyzing 2D fracture trace-line maps. *Comput Geosci.* 2013;60:11–22.
- Zhang C, Han K, Zhang D. Face stability analysis of shallow circular tunnels in cohesive-frictional soils. *Tunn Undergr Space Technol.* 2015;50:345–357.
- Chen J, Zhang D, Huang H, Shadabfar M, Zhou M, Yang T. Image-based segmentation and quantification of weak interlayers in rock tunnel face via deep learning. *Autom Constr.* 2020;120, 103371.
- Huang H-w, Li Q-t, Zhang D-m. Deep learning based image recognition for crack and leakage defects of metro shield tunnel. *Tunn Undergr Space Technol.* 2018;77:166–176.
- Alipour M, Harris DK. Increasing the robustness of material-specific deep learning models for crack detection across different materials. *Eng Struct.* 2020;206, 110157.
- Zhang G, Karakus M, Tang H, Ge Y, Zhang L. A new method estimating the 2D joint roughness coefficient for discontinuity surfaces in rock masses. *Int J Rock Mech Min Sci.* 2014;72:191–198.
- Cui X, Yan E-c. A clustering algorithm based on differential evolution for the identification of rock discontinuity sets. *Int J Rock Mech Min Sci.* 2020;126, 104181.
- Jordá Bordehore L, Riquelme A, Cano M, Tomás R. Comparing manual and remote sensing field discontinuity collection used in kinematic stability assessment of failed rock slopes. *Int J Rock Mech Min Sci.* 2017;97:24–32.
- Reid TR, Harrison JP. A semi-automated methodology for discontinuity trace detection in digital images of rock mass exposures. *Int J Rock Mech Min Sci.* 2000;37:1073–1089.
- Lemy F, Hadjigeorgiou J. Discontinuity trace map construction using photographs of rock exposures. *Int J Rock Mech Min Sci.* 2003;40:903–917.
- Slob S, Van Knapen B, Hack R, Turner K, Kemeny J. Method for automated discontinuity analysis of rock slopes with three-dimensional laser scanning. *Transport Res Rec.* 2005;1913:187–194.
- Zou Q, Cao Y, Li Q, Mao Q, Wang S. CrackTree: automatic crack detection from pavement images. *Pattern Recogn Lett.* 2012;33:227–238.
- Tsai Y, Kaul VK, Mersereau RM. Critical assessment of pavement distress segmentation methods. *J. Transport Eng. Asce.* 2010;136:11–19.
- Abdelqader I, Abudayyeh O, Kelly M. Analysis OF edge-detection techniques for crack identification IN bridges. *J Comput Civ Eng.* 2003;17:255–263.
- Zalama E, Gomezgarciabermejo J, Medina R, Llamas J. Road crack detection using visual features extracted by gabor filters. *Comput Aided Civ Infrastruct Eng.* 2014;29:342–358.
- Reid TR, Harrison JP. Automated tracing of rock mass discontinuities from digital images. *Int J Rock Mech Min Sci.* 1997;34.
- Zhou M, Shadabfar M, Huang H, Leung Y, Uchida S. Meta-modelling of coupled thermo-hydro-mechanical behaviour of hydrate reservoir. *Comput. Geotech.* 2020;128:103848.
- Zhao S, Shadabfar M, Zhang D, Chen J, Huang H. Deep learning-based classification and instance segmentation of leakage-area and scaling images of shield tunnel linings. *Struct. Control Heal. Monit.* 2021;1–22.
- Huang H, Zhao S, Zhang D, Chen J. Deep learning-based instance segmentation of cracks from shield tunnel lining images. *Struct. Infrastruct. Eng.* 2020;1–14.
- Xiao S, Zhang J, Ye J, Zheng J. Establishing region-specific N – Vs relationships through hierarchical Bayesian modeling. *Eng. Geol.* 2021;287(6):1–14.
- Ying L, Salari E. Beamlet transform-based technique for pavement crack detection and classification. *Comput Aided Civ Infrastruct Eng.* 2010;25:572–580.
- Yeum CM, Dyke SJ. Vision-based automated crack detection for bridge inspection. *Comput Aided Civ Infrastruct Eng.* 2015;30:759–770.
- Beckman GH, Polyzois D, Cha Y-J. Deep learning-based automatic volumetric damage quantification using depth camera. *Autom Constr.* 2019;99:114–124.
- Nhat-Duc H, Nguyen Q-L, Tran V-D. Automatic recognition of asphalt pavement cracks using metaheuristic optimized edge detection algorithms and convolution neural network. *Autom Constr.* 2018;94:203–213.
- Li S, Zhao X, Zhou G. Automatic Pixel-Level Multiple Damage Detection of Concrete Structure Using Fully Convolutional Network. Computer-Aided Civil and Infrastructure Engineering; 2019.
- Tao X, Zhang D, Ma W, Liu X, Xu D. Automatic metallic surface defect detection and recognition with convolutional neural networks. *Appl Sci.* 2018;8:1575.
- Long J, Shelhamer E, Darrell T. Fully Convolutional Networks for Semantic Segmentation, the 28th IEEE Conference on Computer Vision and Pattern Recognition. 2015:3431–3440. Boston, MA, USA.
- Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille AL. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans Pattern Anal Mach Intell.* 2018;40:834–848.
- Liang-Chieh Chen GP. Senior Member, IEEE, Iasonas Kokkinos, Member, IEEE, Kevin Murphy, Alan L. Yuille, DeepLab: Semantic Image Segmentation with Deep Convolutional Nets. Atrous Convolution, and Fully Connected CRFs, CV; 2017:1–14.

- 34 Ronneberger O, Fischer P, Brox T. *U-net: Convolutional Networks for Biomedical Image Segmentation*, International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2015:234–241.
- 35 Badrinarayanan V, Kendall A, Cipolla R. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell*. 2017;39:2481–2495.
- 36 Simonyan K, Zisserman A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2014. arXiv preprint arXiv:1409.1556.
- 37 He K, Zhang X, Ren S, Sun J. *Deep Residual Learning for Image Recognition*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:770–778.
- 38 Szegedy C, Liu W, Jia Y, et al. *Going Deeper with Convolutions*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015:1–9.
- 39 He K, Zhang X, Ren S, Sun J. *Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition*. 2014.
- 40 Liang-Chieh Chen YZ, George Papandreou, Florian Schroff, Hartwig Adam, Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. CV; 2018: 1–18.
- 41 Lafferty J, McCallum A, Pereira F. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*, International Conference on Machine Learning. 2001:282–289.
- 42 Xue Y, Li Y. A fast detection method via region-based fully convolutional neural networks for shield tunnel lining defects. *Comput Aided Civ Infrastruct Eng*. 2018;33: 638–654.
- 43 Zhao S, Zhang DM, Huang HW. Deep learning-based image instance segmentation for moisture marks of shield tunnel lining. *Tunn Undergr Space Technol*. 2020;95, 103156.
- 44 Xue Y, Cai X, Shadabfar M, Shao H, Zhang S. Deep learning-based automatic recognition of water leakage area in shield tunnel lining. *Tunn Undergr Space Technol*. 2020;104, 103524.
- 45 Dong Y, Wang J, Wang Z, et al. A deep-learning-based multiple defect detection method for tunnel lining damages. *IEEE Acc*. 2019;7:182643–182657.
- 46 Chen J, Zhou M, Zhang D, Huang H, Zhang F. Quantification of water inflow in rock tunnel faces via convolutional neural network approach. *Autom Construc*. 2021;123, 103526.
- 47 Gu J, Wang Z, Kuen J, et al. Recent advances in convolutional neural networks. *Pattern Recogn*. 2018;77:354–377.
- 48 Zakeri H, Nejad FM, Fahimifar A. Image based techniques for crack detection, classification and quantification in asphalt pavement: a review. *Arch Comput Methods Eng*. 2017;24:935–977.
- 49 Bolkas D, Vazaios I, Peidou A, Vlachopoulos N. Detection of rock discontinuity traces using terrestrial LiDAR data and space-frequency transforms. *Geotech Geol Eng*. 2018; 36:1745–1765.
- 50 Guo J, Liu Y, Wu L, et al. A geometry- and texture-based automatic discontinuity trace extraction method for rock mass point cloud. *Int J Rock Mech Min Sci*. 2019;124, 104132.
- 51 Han S, Wang G, Li M. A trace map comparison algorithm for the discrete fracture network models of rock masses. *Comput Geosci*. 2018;115:31–41.
- 52 Li X, Chen J, Zhu H. A new method for automated discontinuity trace mapping on rock mass 3D surface model. *Comput Geosci*. 2016;89:118–131.
- 53 Ballard DH. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recogn*. 1987;13:714–725.
- 54 Kemeny JM, Post R. Estimating three-dimensional rock discontinuity orientation from digital images of fracture traces. *Comput Geosci*. 2003;29:65–77.
- 55 Wang C, Sha A, Sun Z. *Pavement Crack Classification Based on Chain Code, Fuzzy Systems and Knowledge Discovery*. 2010:593–597.
- 56 Ling W. A region filling algorithm based on crack chain code description. *J. Image Graphics*. 2007;11(12):121–124.
- 57 Jimenez-Rodriguez R, Sitar N. Inference of discontinuity trace length distributions using statistical graphical models. *Int J Rock Mech Min Sci*. 2006;43:877–893.
- 58 Mauldon M, Dunne WM, Rohrbaugh MB. Circular scanlines and circular windows: new tools for characterizing the geometry of fracture traces. *J Struct Geol*. 2001;23: 247–258.
- 59 Zhang P, Zhao Q, Tannant DD, Ji T, Zhu H. 3D mapping of discontinuity traces using fusion of point cloud and image data. *Bull Eng Geol Environ*. 2018;78:2789–2801.
- 60 Zhao Y, Feng Z, Liang W, Yang D, Hu Y, Kang T. Investigation of fractal distribution law for the trace number of random and grouped fractures in a geological mass. *Eng Geol*. 2009;109:224–229.
- 61 Zhang L, Einstein HH. Estimating the intensity of rock discontinuities. *Int J Rock Mech Min Sci*. 2000;37:819–837.
- 62 Zhang L, Einstein HH. Estimating the mean trace length of rock discontinuities. *Rock Mech Rock Eng*. 1998;31:217–235.
- 63 Priest SD, Hudson JA. Estimation OF discontinuity spacing and trace length using SCANLINE surveys. *Int J Rock Mech Min Sci Geomech Abstr*. 1981;18:183–197.
- 64 Cruden DM. Describing the size of discontinuities. *Int J Rock Mech Min Sci Geomech Abstr*. 1977;14:133–137.
- 65 He K, Sun J, Tang X. *Guided Image Filtering*, European Conference on Computer Vision. 2010:1–14.
- 66 Liu Y, Yao J, Lu X, Xie R, Li L. DeepCrack: a deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing*. 2019;338:139–153.
- 67 Darlington J, Field AJ, Harrison PG, et al. *Parallel Programming Using Skeleton Functions*, International Conference on Parallel Architectures and Languages Europe. Springer; 1993:146–160.
- 68 Felkel P, Obdrzalek S. *Straight Skeleton Implementation*, Proceedings of Spring Conference on Computer Graphics. 1998. Citeseer.
- 69 Rehman H, Naji AM, Kim J-j, Yoo H. Extension of tunneling quality index and rock mass rating systems for tunnel support design through back calculations in highly stressed jointed rock mass: an empirical approach based on tunneling data from Himalaya. *Tunn Undergr Space Technol*. 2019;85:29–42.
- 70 Zhang Q, Huang X, Zhu H, Li J. Quantitative assessments of the correlations between rock mass rating (RMR) and geological strength index (GSI). *Tunn Undergr Space Technol*. 2019;83:73–81.
- 71 Warren SN, Kallu RR, Barnard CK. Correlation of the rock mass rating (RMR) system with the unified soil classification system (USCS): introduction of the weak rock mass rating system (W-RMR). *Rock Mech Rock Eng*. 2016;49:4507–4518.
- 72 Z. Bieniawski, The Rock Mass Rating (RMR) System (Geomechanics Classification) in Engineering Practice, Rock Classification Systems for Engineering Purposes, ASTM International1988.
- 73 Perez L, Wang J. *The Effectiveness of Data Augmentation in Image Classification Using Deep Learning*, arXiv Preprint arXiv:1712.04621. 2017.
- 74 Huang H, Cheng W, Zhou M, Chen J, Zhao S. Towards automated 3D inspection of water leakages in shield tunnel linings using mobile laser scanning data. *Sensors*. 2020;20:6669.