ORIGINAL PAPER

OCM: an intelligent recognition method of rock discontinuity based on optimal color mapping of 3D Point cloud via deep learning

Keshen Zhang[1] · Wei Wu[1,2,3] · Yongsheng Liu[4] · Yong Huang[5] · Min Zhang[6] · Hehua Zhu[1,2,3]

Abstract

Discontinuities largely influence the mechanical properties of rock joints. However, traditional discontinuity recognition methods often require manual intervention during processing. This paper proposes a new deep-learning-based method for discontinuity recognition using 3D point clouds. A neighborhood PCA-weighted oriented contraction method is proposed to extract point cloud skeletons as discontinuity intersection lines. Then an optimal color mapping (OCM) method is proposed to establish the optimal mapping relationship between 3D normal vectors and RGB, converting 3D point clouds to 2D OCM images for discontinuity segmentation. The convolutional neural network of Mask R-CNN is adopted to segment discontinuities from OCM images. Finally, 3D discontinuities can be generated from discontinuity-segmented OCM images. Forty-two rock slope image sequences and a rock slope point cloud are collected and labeled, generating 4632 OCM images including 430,613 discontinuity planes after data augmentation for training. Three cases of rock slopes and rock tunnel excavation faces are adopted for testing. The average recognition time per 3D point cloud model is approximately 12 s due to the high recognition efficiency of Mask R-CNN for 2D images. The results show the proposed method can recognize discontinuities close to manual judgements with high accuracy, good robustness to point cloud density variations, and good adaptability to different rock engineering scenarios.

Highlights

- An NPW-OC method is proposed to extract point cloud skeletons.
- An OCM method is proposed to assign 3D normal vectors with optimal RGB.
- OCM images are generated to assign discontinuities with different and uniform color.
- Deep-learning-based method is used for intelligent recognition of discontinuities.
- Conversion of discontinuity recognition from 3D point clouds to 2D OCM images.
- The results show good efficiency, accuracy, robustness, and adaptability .

✉ Wei Wu
weiwu@tongji.edu.cn

1 College of Civil Engineering, Tongji University, Shanghai 200092, China

2 Key Laboratory of Geotechnical and Underground Engineering of Ministry of Education, Tongji University, Shanghai 200092, China

3 State Key Laboratory of Disaster Reduction in Civil Engineering, Tongji University, Shanghai 200092, China

4 China Railway Tunnel Group Co., Ltd, Guangzhou 511458, China

5 China Railway First Survey and Design Institute Group Co., Ltd, Xian 710043, China

6 Shudao Investment Group Co., Ltd, Chengdu 610095, China

1 Introduction

Discontinuities largely influence the mechanical behaviors of rock joints (Barton 1978). The international society for rock mechanics (ISRM) proposed ten parameters to quantitatively describe the properties of rock discontinuities, including orientation, roughness, aperture, wall strength, filling, seepage, spacing, persistence, number of sets, and block size (Barton 1978). In addition, the geometry of discontinuities has been applied to a variety of practices, including the hazard identification and monitoring (Herrera et al. 2010; Jones and Hobbs 2021), structural geology (Cawood et al. 2017), landslides (Jaboyedoff et al. 2012), earthquakes (Rathje and Franke 2016), identification of fault areas (Chen et al. 2015), mapping ground texture (Yan et al. 2015), and geospatial analysis of data (Deibe et al. 2020; de Oliveira et al. 2021; Kong 2021; Smith and Holden 2021). Therefore, accurate recognition of discontinuities is essential for the analysis of rock engineering.

The traditional method relies on the manual visual recognition of discontinuity planes and contact measurements of engineers to collect geometry data, which is time-consuming and easily affected by user bias (Priest 1993; Abellán et al. 2014; Gigli et al. 2022). Alternatively, remote sensing methods such as laser scanning and stereophotogrammetry have been adopted to collect high-resolution 3D point clouds for discontinuity recognition (Ferrero et al. 2016; Chen et al. 2017; Giordan et al. 2018; Li et al. 2019). Therefore, various methods based on 3D point clouds have been proposed for discontinuity recognition (Daghigh et al. 2022). These methods can be mainly divided into two types.

The first type of discontinuity recognition method requires carrying out orientation grouping in advance, and the discontinuity planes are then extracted based on the optimal orientation grouping results. For instance, Riquelme et al. (2014) adopted a kernel density estimation (KDE) to recognize the optimal discontinuity sets using two manually adjusted parameters of cone filter and max poles filter. Then the density-based scan algorithm with noise (DBSCAN) method was used to generate independent discontinuities with two user-defined parameters of ε and min-pts. Kong et al. (2020) adopted the clustering by fast searches an finding density peaks (CFSFDP) (Rodriguez and Laio 2014) method to perform the optimal orientation grouping using a user-defined parameter of cutoff distance. The discontinuity planes were then recognized using DBSCAN. However, these are semi-automatic methods requiring manual adjustment of parameters when dealing with different rock models. In addition, there are automatic methods using clustering validity indexes, such as Silhouette index (Rousseeuw 1987; Chen et al. 2016; Ruiz Pereira et al. 2021),

Caliński–Harabasz index (Caliński and Harabasz 1974; Esmaeilzadeh and Shahriar 2019; Singh et al. 2022a) and Xie–Beny index (Xie and Beni 1991; Hammah and Curran 2000; Liu et al. 2022; Yan et al. 2022), etc., for extracting optimal group number and grouping results. However, the discontinuity recognition accuracy is heavily dependent on the effects of optimal grouping results which can be inconsistent among different validity indexes because of their different calculation emphasis.

The second type of method is designed to directly extract discontinuity planes without grouping in advance. For example, the region-growing method is used to extract the discontinuity planes based on the variation of normals or curvatures in local surfaces (Wang et al. 2017; Ge et al. 2018). However, the growing seed location and growing criteria require to be predefined and modified for different datasets (Vo et al. 2015). The computational efficiency is also limited because of the iterative growing process. Random sample consensus (RANSAC) method is also used to extract discontinuity planes by iterative estimation of plane parameters (Raguram et al. 2013). However, a threshold to judge whether a point is within a plane is required to be manually fine-tuned for different datasets. The computational cost is still high because of the iterative nature and the randomness process (Raguram et al. 2013; Liu et al. 2019). In addition, the amplitude and phase decomposition (APD) approach is also proposed for discontinuity plane detection with the fast Fourier transform (FFT)-based decomposition of signals (Singh et al. 2022a). However, this method also requires manual selection of several parameters within the optimal range for different datasets.

In addition, compared with the traditional machine learning methods, the rapid development of deep learning techniques and artificial intelligence gives another solution to improve the accuracy, efficiency, and automation in discontinuity plane recognition. As one of the representative techniques in computer vision in deep learning field, the convolution neural network (CNN) has been widely used in many aspects of rock engineering, such as rock type classification (Ran et al. 2019; Alférez et al. 2021; Chen et al. 2023), rock discontinuity trace detection (Chen et al. 2021; Qiao et al. 2022), rock strength prediction (Zhao et al. 2023), etc. However, because of the lack of large-scale 3D rock discontinuity data with various rock types and the high labeling expense of 3D discontinuity planes, the deep-learning-based method of 3D discontinuity plane recognition has not been widely applied.

This paper proposes an intelligent recognition method of rock discontinuity based on optimal color mapping (OCM) of 3D point clouds via deep learning. The proposed method mainly consists of five steps, including step 1 of point cloud preprocessing and discontinuity intersection line extraction, step 2 of normal vector OCM, step 3 of generation of

OCM images, step 4 of data preparation and Mask R-CNN model training, and step 5 of discontinuity recognition by the trained Mask R-CNN model. This paper is organized as follows: an introduction of rock discontinuity recognition methods using 3D point clouds is presented in Sect. 1, the specific procedure of the proposed method is presented in Sect. 2, the proposed method is applied to three cases in Sect. 3, the method is discussed in Sect. 4 and some conclusions are drawn in Sect. 5.

2 Methodology

This paper proposes an intelligent recognition method of rock discontinuities based on OCM of 3D point clouds via deep learning. The detailed flow chart of the proposed method is shown in Fig. 1. This method starts with the input of 3D point cloud models and is mainly composed of five steps. In step 1, after obtaining 3D point clouds of rock mass, a neighborhood PCA-weighted oriented contraction (NPW-OC) method is proposed to extract sharp point skeletons as discontinuity intersection lines. In step 2, the OCM method is proposed to map normal vectors to optimal RGB colors. In step 3, the color-mapped point cloud combined with sharp point skeletons is used for OCM image generation. In step 4, OCM images are manually labeled with ground truth discontinuities and augmented. Next comes a two-stage operation. In the training stage of step 4, the Mask R-CNN model is adopted for training with augmented OCM images and the mask images corresponding to labeled OCM images. In the recognition stage of step 5, discontinuities are segmented by the trained Mask R-CNN model using OCM images to generate mask images. Finally in step 5, 3D discontinuities are mapped from 3D point cloud models based on the mask images of segmented discontinuities.

A rock slope case is adopted to illustrate each step of the proposed method. The rock slope is located in Mountain Lao, Qingdao, China. An Iphone12 mobile phone is used to take seven images (4032×3024) at different locations in front of the slope (Fig. 2a). The image sequence is then processed by the Meshroom opensource software to reconstruct the 3D point cloud model. The region of interest (ROI) is shown in the black rectangle in Fig. 2b, which contains 397,546 points with the approximate average spacing of adjacent points of 2.20 cm.

2.1 Point cloud preprocessing and discontinuity intersection line extraction (step 1)

After obtaining the raw point cloud, the preprocessing is first to be performed. Considering the intersection lines of adjacent discontinuity planes are commonly used for discontinuity segmentation (Khaloo and Lattanzi 2017;

Li et al. 2016; Singh et al. 2022b), the Laplacian-based contraction method is used for the extraction of point cloud skeletons located on the intersection lines of adjacent discontinuities (Zhang et al. 2020). However, this method requires point cloud meshing and cannot be directly performed on raw point clouds. Therefore, this section proposes a neighborhood principle-component-analysis (PCA)-weighted oriented contraction (NPW-OC) method to extract intersection line as point cloud skeletons from raw point clouds without meshing.

2.1.1 Normal vector calculation and hemispherization (step 1.1)

After obtaining 3D point cloud models, normal vectors are required to be calculated first. The least square method and the PCA are often used in normal vector estimation (Sturzenegger and Stead 2009; Abellán et al. 2014). In addition, there are some adaptive methods to improve the robustness of normal vector quality to noises (Wang et al. 2013). In this paper, the PCA method is adopted for normal vector calculation.

Given the point cloud $P = \{p_1, p_2, \ldots, p_N\}$ (N denotes the point number), then the normal vector of a point $p_0 \in P$ requires to calculate the covariance matrix as

$$M_{cov} = \frac{1}{k_{nn}} \sum_{i=1}^{k_{nn}} (p_i - p_0)(p_i - p_0)^T = \sum_{i=1}^{3} \lambda_i \overrightarrow{e_i} \overrightarrow{e_i}^T \tag{1}$$

where p_i is the ith point of k_{nn} nearest points of p_0 with Euclidean distance. $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are the eigenvalues and the normal vector vec_0 of p_0 is the 3rd eigenvector $\overrightarrow{e_3}$ of M_{cov}. Considering that small k_{nn} (e.g., knn=15) can cause significant noise in normal vector calculation and large k_{nn} (e.g., k_{nn}>30) can significantly smooth local curvatures (Riquelme et al. 2014), $k_{nn} = 20$ is set as an initial value in this paper. Equation (1) is programmed from scratch in Matlab.

Because normal vector hemispherization is commonly used in discontinuity analysis (Jimenez-Rodriguez and Sitar 2006), all vectors with z coordinates smaller than zero are reversed to the upper unit hemisphere.

2.1.2 Sharp point detection (step 1.2)

Sharp points are referred to edge points and corner points located in large curvatures (Wang et al. 2013). Therefore, the neighbor angle variation is adopted for the sharp point detection.

The distance metric is first defined as the acute angle of normal vectors (Jimenez-Rodriguez and Sitar 2006).
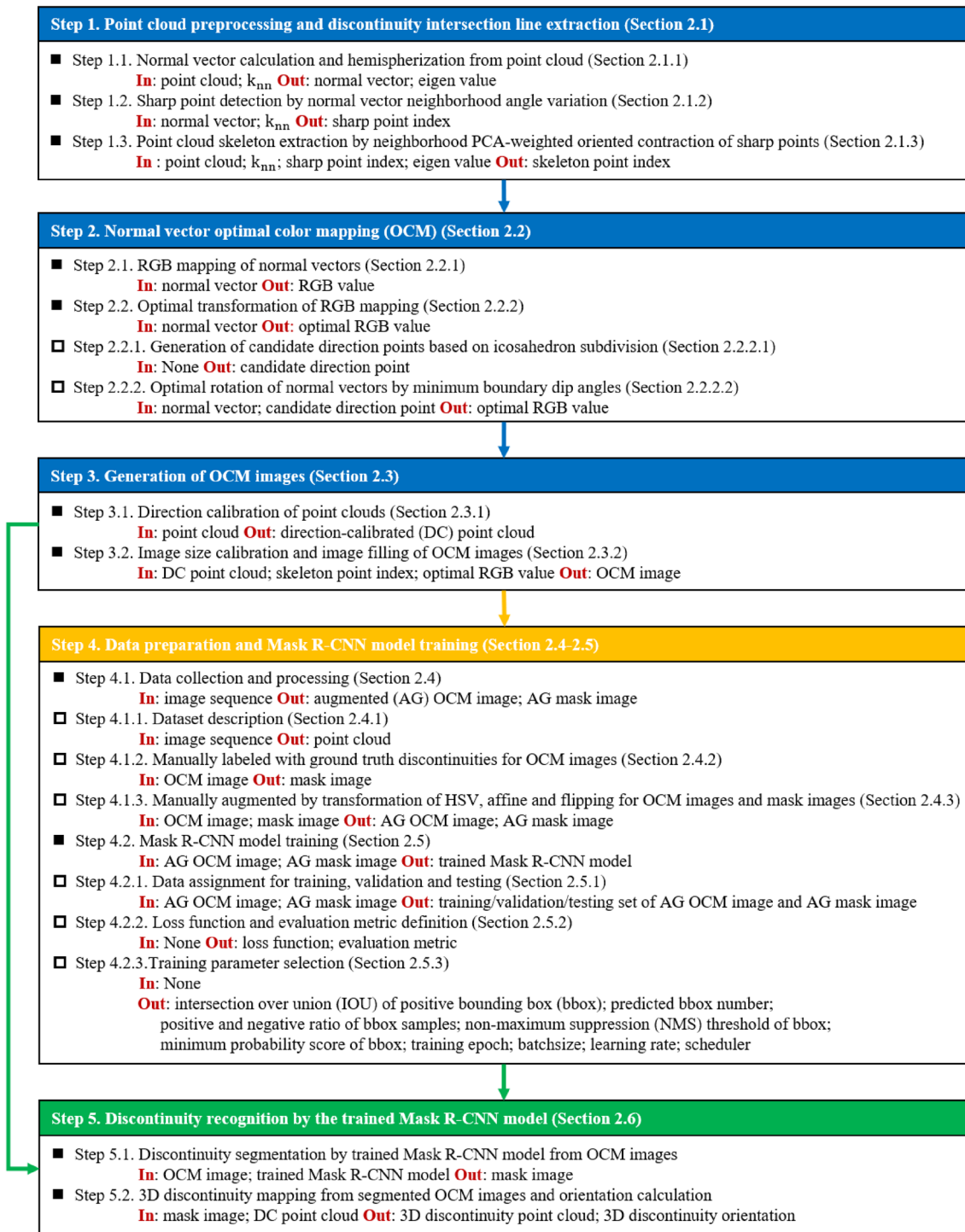
Step 1. Point cloud preprocessing and discontinuity intersection line extraction (Section 2.1)

- Step 1.1. Normal vector calculation and hemispherization from point cloud (Section 2.1.1)
  In: point cloud; k_{nn} **Out**: normal vector; eigen value
- Step 1.2. Sharp point detection by normal vector neighborhood angle variation (Section 2.1.2)
  In: normal vector; k_{nn} **Out**: sharp point index
- Step 1.3. Point cloud skeleton extraction by neighborhood PCA-weighted oriented contraction of sharp points (Section 2.1.3)
  In : point cloud; k_{nn}; sharp point index; eigen value **Out**: skeleton point index

Step 2. Normal vector optimal color mapping (OCM) (Section 2.2)

- Step 2.1. RGB mapping of normal vectors (Section 2.2.1)
  In: normal vector **Out**: RGB value
- Step 2.2. Optimal transformation of RGB mapping (Section 2.2.2)
  In: normal vector **Out**: optimal RGB value
- ☐ Step 2.2.1. Generation of candidate direction points based on icosahedron subdivision (Section 2.2.2.1)
  In: None **Out**: candidate direction point
- ☐ Step 2.2.2. Optimal rotation of normal vectors by minimum boundary dip angles (Section 2.2.2.2)
  In: normal vector; candidate direction point **Out**: optimal RGB value

Step 3. Generation of OCM images (Section 2.3)

- Step 3.1. Direction calibration of point clouds (Section 2.3.1)
  In: point cloud **Out**: direction-calibrated (DC) point cloud
- Step 3.2. Image size calibration and image filling of OCM images (Section 2.3.2)
  In: DC point cloud; skeleton point index; optimal RGB value **Out**: OCM image

Step 4. Data preparation and Mask R-CNN model training (Section 2.4–2.5)

- Step 4.1. Data collection and processing (Section 2.4)
  In: image sequence **Out**: augmented (AG) OCM image; AG mask image
- ☐ Step 4.1.1. Dataset description (Section 2.4.1)
  In: image sequence **Out**: point cloud
- ☐ Step 4.1.2. Manually labeled with ground truth discontinuities for OCM images (Section 2.4.2)
  In: OCM image **Out**: mask image
- ☐ Step 4.1.3. Manually augmented by transformation of HSV, affine and flipping for OCM images and mask images (Section 2.4.3)
  In: OCM image; mask image **Out**: AG OCM image; AG mask image
- Step 4.2. Mask R-CNN model training (Section 2.5)
  In: AG OCM image; AG mask image **Out**: trained Mask R-CNN model
- ☐ Step 4.2.1. Data assignment for training, validation and testing (Section 2.5.1)
  In: AG OCM image; AG mask image **Out**: training/validation/testing set of AG OCM image and AG mask image
- ☐ Step 4.2.2. Loss function and evaluation metric definition (Section 2.5.2)
  In: None **Out**: loss function; evaluation metric
- ☐ Step 4.2.3.Training parameter selection (Section 2.5.3)
  In: None
  Out: intersection over union (IOU) of positive bounding box (bbox); predicted bbox number;
  positive and negative ratio of bbox samples; non-maximum suppression (NMS) threshold of bbox;
  minimum probability score of bbox; training epoch; batchsize; learning rate; scheduler

Step 5. Discontinuity recognition by the trained Mask R-CNN model (Section 2.6)

- Step 5.1. Discontinuity segmentation by trained Mask R-CNN model from OCM images
  In: OCM image; trained Mask R-CNN model **Out**: mask image
- Step 5.2. 3D discontinuity mapping from segmented OCM images and orientation calculation
  In: mask image; DC point cloud **Out**: 3D discontinuity point cloud; 3D discontinuity orientation

Fig. 1 Flow chart of the proposed method. In and Out denote the input and output data types of each step, respectively

Given the normal vector set $Vec = \{vec_1, vec_2, \ldots, vec_N\}$, the distance metric is defined as

$$dist(vec_i, vec_j) = arccos\left(\frac{|vec_i \cdot vec_j|}{|vec_i||vec_j|}\right) \tag{2}$$

Fig. 2 Data collection and processing of a rock slope. **a** Image sequence. ROI is in the black rectangle region. **b** 3D reconstructed point clouds
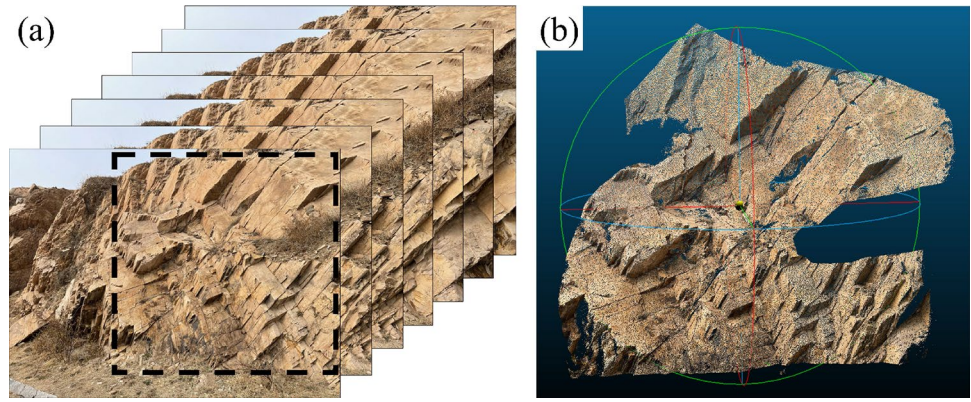
where vec_i and vec_j denote any two normal vectors. All the arccos calculation in this paper is performed using the *acosd* function in Matlab.

Then the neighbor variation angle δ_i is defined as

$$\delta_i = \frac{1}{k_{nn}} \sum_{j=1}^{k_{nn}} dist(vec_i, vec_j) \tag{3}$$

where vec_j denotes the normal vector of the jth k_{nn} nearest points of p_i. The k-nearest searching algorithm is performed using the *knnsearch* function in Matlab.

Finally, the sharp point set Set_{shp} are defined as

$$Set_{shp} = \{i | \delta_i > \frac{1}{N} \sum_{j=1}^{N} \delta_j, i \in \{1, 2, \dots, N\}\} \tag{4}$$

Equation (4) is performed using the *find* function in Matlab.

2.1.3 Neighborhood PCA-weighted oriented contraction (step 1.3)

The diversity of rock mass and point cloud density can lead to over sparse sharp points near intersection lines in Sect. 2.1.2, which occupies a large area of discontinuities. The uneven distribution of sharp points can also reduce the continuity of intersection lines. Thus, the point cloud contraction algorithm is considered to extract sharp point skeletons as intersection lines. However, traditional point cloud contraction algorithms often require meshing and cannot be directly performed on raw point clouds (Au et al. 2008; Cao et al. 2010; Zhang et al. 2020). Therefore, a NPW-OC method is proposed to achieve the oriented contraction of sharp points without meshing. Figure 3b shows the point cloud contraction skeleton of Fig. 3a by the proposed method.

Considering the eigenvalue of PCA indicates the dispersion degree of the neighbor point distribution along eigenvector directions (Lee et al. 2006), the eigenvector $\vec{e_1}$ corresponding to the largest eigenvalue λ_1 is, therefore, used to represent the local tangent direction of the point cloud (shown in Figs. 3d and 4). To evaluate the dominance of tangent directions, a linear significance parameter u_1 is defined as

$$u_1 = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} \tag{5}$$

In general, the points (p_i in Fig. 5a) far from the skeleton can result in a larger k_{nn} neighbor point distribution length (similar to u_1) along $\vec{e_1}$ than the points (p_{cen} in Fig. 5a) closer to the skeleton. Therefore, a parameter w_c is defined by u_1 to give points near skeletons large contraction weights as

$$w_c = u_1^2 \tag{6}$$

Figure 3e shows the value of w_c. It can be seen that w_c is smaller at the sharp point away from the skeleton and larger near the skeleton as expected. Then the NPW-OC is performed using w_c to make sharp points move toward the skeleton. Given a point p_i in the point cloud, its weighted contraction point $p_i\prime$ is defined as

$$p_i' = \frac{1}{\sum_{j=1}^{k_{nn}} w_{cj}} \sum_{j=1}^{k_{nn}} w_{cj}(p_{ij} - p_i) \tag{7}$$

where p_{ij} denotes the jth the nearest point of p_i. $p_i\prime$ is denoted in Fig. 5a.

To ensure the continuity of the contracted sharp points along the skeleton, sharp points are designed to move perpendicular to the tangent directions. Therefore, the displacement vector $\overrightarrow{p_ip_i\prime}$ is projected on $\vec{e_2}$ to generate the orientation calibrated point p_i'' as
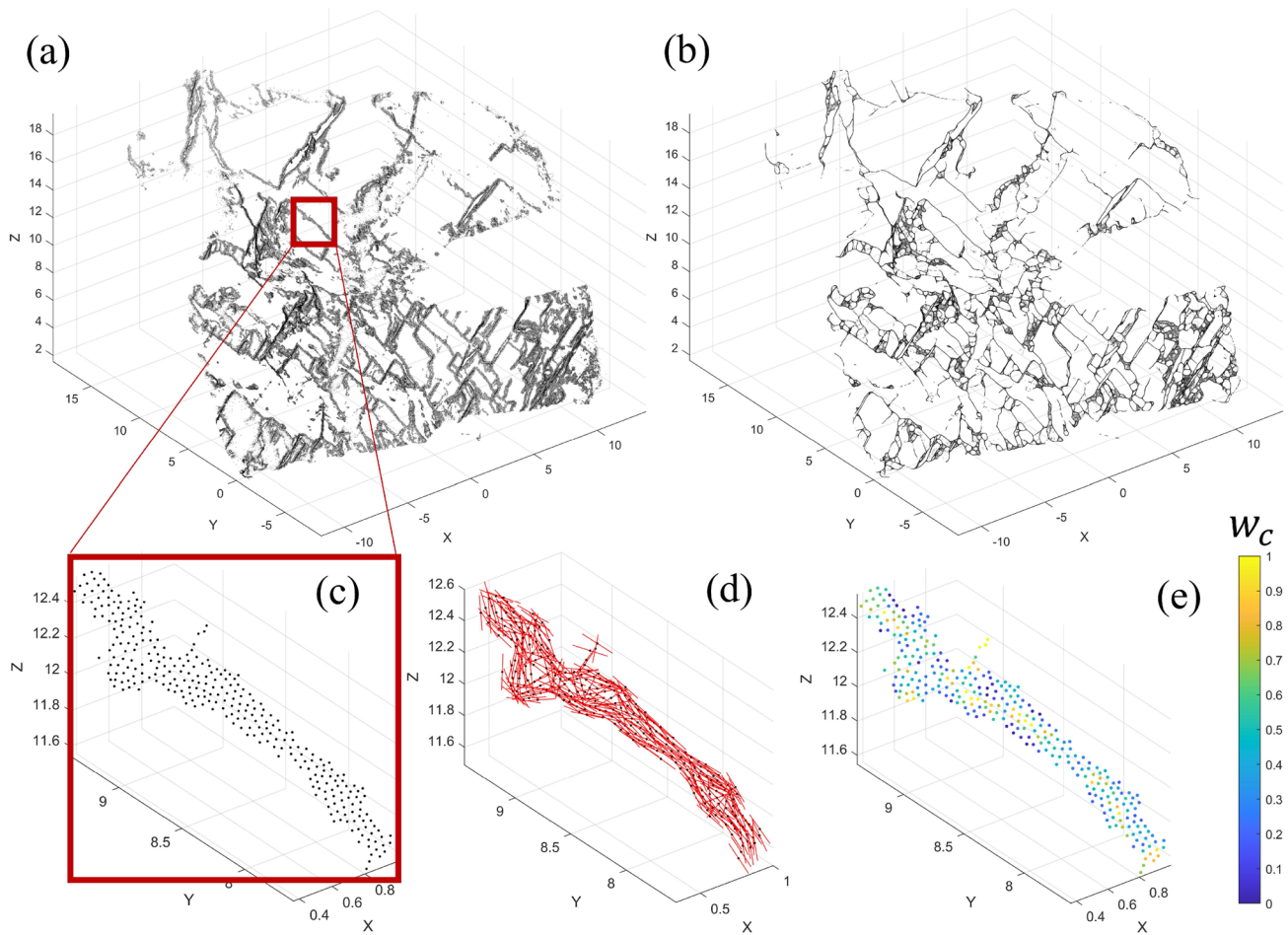
Fig. 3 Process of the neighborhood PCA-weighted oriented contraction (NPW-OC). **a** Sharp points. **b** The point cloud skeleton of sharp points after NPW-OC. **c** An example set of sharp points. **d** Tangent vectors of the example sharp points. **e** w_c of example sharp points
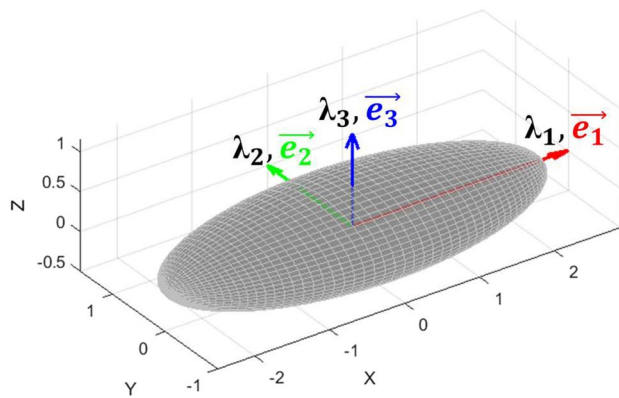

Fig. 4 Eigen values and eigen vectors of PCA

$$p_i'' = \left(p_i' - p_i\right) \frac{\left|\vec{e_2}\left(p_i' - p_i\right)\right|}{\left\|p_i' - p_i\right\|_2 \left\|\vec{e_2}\right\|_2} + p_i \tag{8}$$

where $\|\cdot\|_2$ denotes the mode length. The oriented contracted point p_i'' is shown in Fig. 5a.

All the sharp points are performed by the NPW-OC method based on Eqs. (5) to (8) which are programmed from scratch in Matlab. Figure 5a shows the skeleton generated by one-time contraction. It can be seen in Fig. 5c–d that more contraction times can generate thinner and more accurate skeletons. However, large contraction times can also reduce the computational efficiency. Considering the aim of contraction is to improve the accuracy of discontinuity intersection lines without quantitative requirements, two-time contraction is used as the initial NPW-OC time with the balance between accuracy and efficiency.
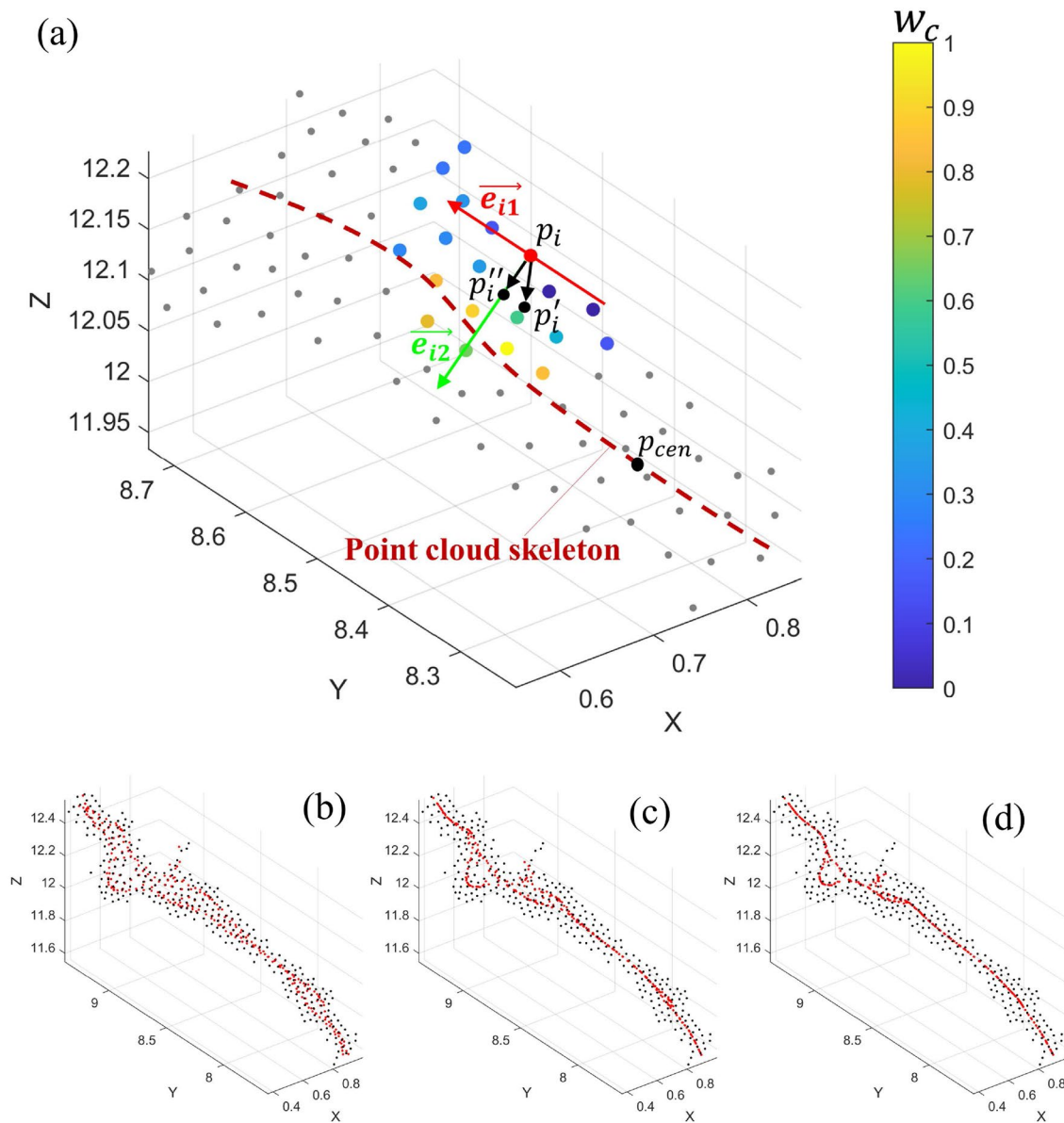
Fig. 5 Explanation of NPW-OC process. **a** Point cloud contraction process. **b** Contraction results of one-time NPW-OC, **c** Contraction results of two-time NPW-OC. **d** Contraction results of three-time NPW-OC

2.2 Normal vector optimal color mapping (step 2)

After obtaining normal vectors, the philosophy of coloring with normal vectors implemented by Jaboyedoff et al. (2007) can be used to assign colors according to the dip and dip direction, which can effectively contribute to the structural analysis of rock mass. For the purpose of further improving the quality of normal vector color mapping to make it more effective and stable for the segmentation of discontinuity planes based on their normal vector colors

from the point cloud models, we propose an OCM method of normal vectors in this section.

To assign colors to normal vectors, the stereographic projection plane of normal vectors is first mapped to HSV color space and then mapped to the RGB space. In addition, considering the boundary points (Fig. 6a) on the stereographic projection plane can cause large difference of colors in the same discontinuity plane (Fig. 6b), an optimal stereographic projection method based on minimum
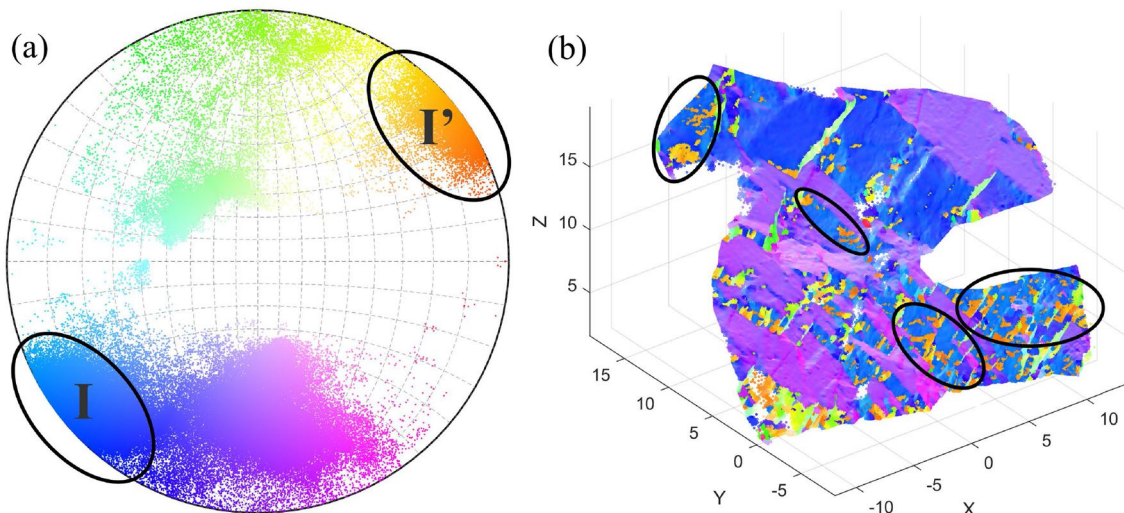
Fig. 6 RGB color mapping results. **a** RGB color mapping of normal vectors. **b** 3D RGB point clouds after color mapping of normal vectors (color figure online)

boundary dip angles (MBDA) is proposed to make the colors within a single discontinuity as uniform as possible.

2.2.1 RGB mapping of normal vectors (step 2.1)

Hemisphere projection of discontinuity poles (such as discontinuity normal vectors) is often used for the description of orientation distribution (Priest 1985; Jimenez-Rodriguez and Sitar 2006). Therefore, normal vectors are first projected to the upper unit hemisphere.

Normal vectors are then mapped to the HSV space. HSV space is one of the most classical color spaces. Since the HSV space is conical, the one-to-one mapping of normal vectors to HSV values is achieved by setting the stereographic projection plane onto the HSV horizontal section. HSV is represented by $H \in [0°, 360°]$, saturation $S \in [0, 1]$, and value $V \in [0, 1]$. To make the color more distinguishable, the stereographic projection plane is set to coincide with the HSV horizontal section of $V = 1$. Then normal vectors are mapped based on the relationship between the stereographic projection plane and the HSV space.

Specifically, given the normalized normal vector set $Vec = \{vec_1, vec_2, \ldots, vec_N\}$, the coordinate of each $vec_i \in Vec$ is $vec_i = [x_i, y_i, z_i]^T$, then except for *Value* = 1, H and S are defined as

$$H = \begin{cases} \frac{1}{360} \arccos\left(\frac{x_i}{\sqrt{x_i^2 + y_i^2}}\right)°, & y_i > 0 \\ 1 - \frac{1}{360} \arccos\left(\frac{x_i}{\sqrt{x_i^2 + y_i^2}}\right)°, & y_i \leq 0 \end{cases} \quad (9)$$

$$S = \sqrt{x_i^2 + y_i^2} \quad (10)$$

Equations (9) and (10) are programmed from scratch in Matlab.

Given the HSV value of vec_i as $[H, S, V]$, the $[R, G, B]$ value is mapped as follows (Smith 1978):

1. $H = 6 * H$
2. $I = floor(H), F = H - I$
3. $M = V \times (1 - S), N = V \times (1 - S \times F), K = V \times (1 - S \times (1 - F))$
4. $[R, G, B] = \begin{cases} [V, K, M], & \text{if } I = 0 \\ [N, V, M], & \text{if } I = 1 \\ [M, V, K], & \text{if } I = 2 \\ [M, N, V], & \text{if } I = 3 \\ [K, M, V], & \text{if } I = 4 \\ [V, M, N], & \text{if } I = 5 \end{cases}$

In step 2, *floor(x)* denotes the integer just less than or equal to x. Figure 6a denotes the normal vector stereographic projection after RGB mapping, and Fig. 6b denotes the corresponding 3D RGB point cloud. The *floor(x)* calculation in step 2 is performed using the *floor* function in Matlab, and other steps are programmed from scratch in Matlab.

2.2.2 Optimal transformation of RGB mapping (step 2.2)

Hemisphere projection can cause normal vectors with approximate 90° dip angles to generate large differences of dip directions near the boundary of the stereographic projection plane. For example, as shown in Fig. 6a, normal vectors in region I and I′ have similar directions, but the

hemispherical projection causes them to be distributed on both sides of the stereographic projection plane, resulting in excessive color differences. This can cause the point color in the same discontinuity plane non-uniform (Fig. 6b), disturbing the color-based segmentation of discontinuities. Therefore, an optimal transformation of normal vectors is proposed to make the hemisphere projection of normal vectors away from the boundary of the stereographic projection plane as far as possible, making point colors in the same discontinuity as uniform as possible.

Table 1 CDPs with different subdivision numbers

Subdivision number n_{div}	2	3	4	5
CDP number	26	91	341	1321
Mean adjacent angle (°)	19.63	12.62	7.21	3.87

2.2.2.1 Generation of candidate direction points (CDPs) based on ortho-icosahedron subdivision (step 2.2.1) Because the optimal transformation of a specific set of normal vectors is unknown in advance, the candidate direction points (CDPs) are proposed to serve as possible rotation directions in the 3D normal vector space. Then the set of normal vectors are rotated according to each of the CDPs' directions to find one of the CDPs as the optimal direction for the RGB mapping of normal vectors according to the proposed method of minimum boundary dip angles. To distribute CDPs uniformly over the entire normal vector space, the ortho-icosahedron subdivision is used to generate CDPs approximately uniformly distributed over the upper unit hemisphere. Therefore, CDPs are generated using the method of Fekete and Treinish (1990) as

1. Input the initial 20 vertices $Vtx = [vtx_1, vtx_2, \ldots, vtx_{12}]$ and 20 triangular patches $Pth = [pth_1, pth_2, \ldots, pth_{20}]$ of the ortho icosahedron (Fig. 7a).
2. For each $pth_i \in Pth$, calculate and normalize the midpoints $Vtx_{add} = [vtx_{i1}, vtx_{i2}, vtx_{i3}]$ of each edge of pth_i, then four new triangular patches are generated as $pth_{add} = [pth_{i1}, pth_{i2}, pth_{i3}]$ (Fig. 7b).
3. Replace pth_i in Pth by Pth_{add}, and merge Vtx_{add} in Vtx.
4. Repeat step 2 to step 3 n_{div} times to generate appropriate number of CDPs.
5. All the points in Vtx and on the upper unit hemisphere are selected as CDPs.

The subdivision time n_{div} decides the number and the accuracy (the mean angle between adjacent CDPs) of CDPs. As shown in Table 1, the more the CDPs, the higher the accuracy. However, excessive numbers of CDPs can reduce the search efficiency of the optimal CDPs. Considering ISRM recommends 5° as the manual measurement error of orientation (Barton 1978), 1321
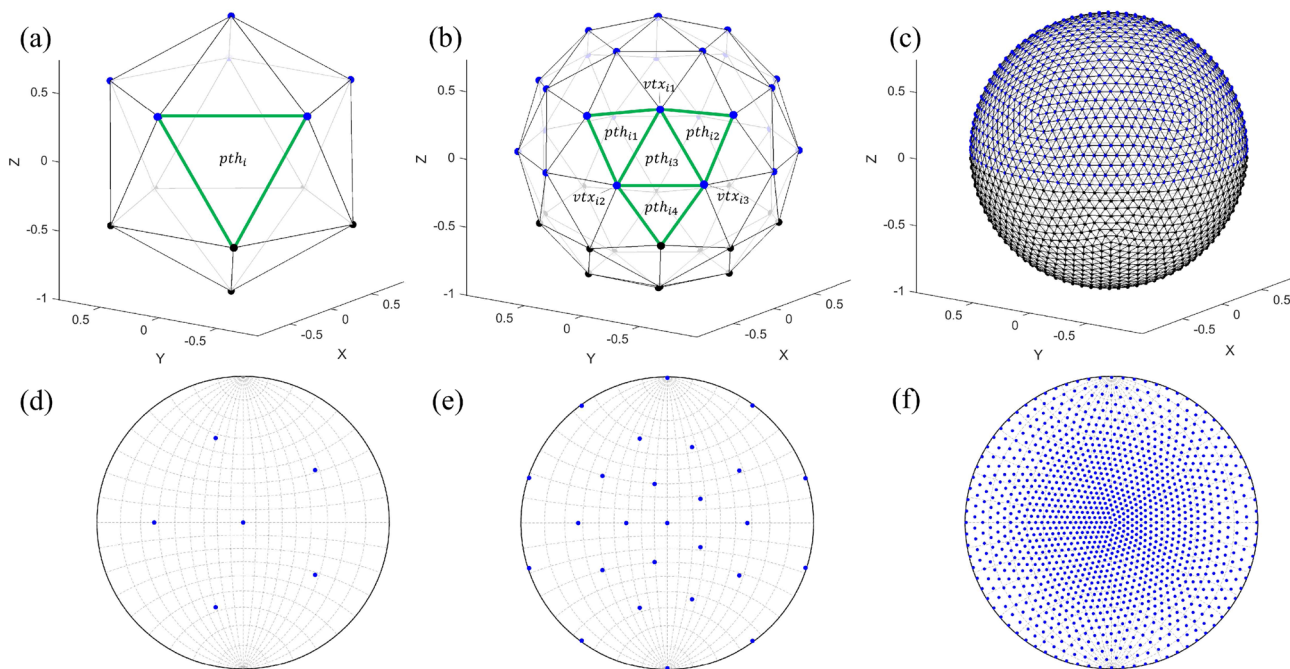
Fig. 7 Subdivision of ortho-icosahedron for CDP generation. **a** Initial vertices and triangular patches of the ortho-icosahedron. **b** Ortho-icosahedron subdivision with $n_{div} = 1$. **c** Ortho-icosahedron subdivision with $n_{div} = 5$. **d** Stereographic projection of initial CDP. **e** Stereographic projection of CDP with $n_{div} = 1$. **f** Stereographic projection of initial CDP with $n_{div} = 5$

CDPs of $n_{div} = 5$ is selected as the default CDPs. The mean adjacent angle is 3.87° less than 5°.

The methods in this section are programmed from scratch in Matlab.

2.2.2.2 Optimal rotation of normal vectors based on minimum boundary dip angles (step 2.2.2)

Normal vectors are required to be distributed away from the boundary of the stereographic projection plane as far as possible to avoid the non-uniform colors in the same discontinuity plane (Fig. 7a, b). Therefore, a method of minimum boundary dip angle is proposed.

Given normal vector set $Vec = \{vec_1, vec_2, \ldots, vec_N\}$, CDP set $Vtx = \{vtx_1, vtx_2, \ldots, vtx_M\}$ (M denotes CDP number), the proposed method is performed as follows:

1. According to ISRM's recommendation to define 5° as the manual measurement error of orientations, normal vectors with dip angles larger than 85° are selected as boundary vectors.

2. Given a CDP $vtx_i \in Vtx$ and the coordinate $vtx_i = [x_i, y_i, z_i]^T$, calculate the rotation matrix Rot_i to rotate vtx_i to $[0, 0, 1]^T$. First calculate the angle ang_z that rotates vtx_1 clockwise around the z-axis to the positive x-axis as

$$ang_z = \begin{cases} 360° - \arccos\left(\dfrac{x_i}{\sqrt{x_i^2+y_i^2}}\right)^\circ, & y_i > 0 \\[3mm] \arccos\left(\dfrac{x_i}{\sqrt{x_i^2+y_i^2}}\right)^\circ, & y_i \le 0 \end{cases} \quad (11)$$

Then calculate the angle ang_y that rotates vtx_i clockwise around the y-axis to the positive z-axis as

$$ang_y = -\arccos\left(\dfrac{z_i}{\sqrt{x_i^2 + y_i^2 + z_i^2}}\right)^\circ \quad (12)$$

Therefore, Rot_i is defined as

$$Rot_i = \begin{bmatrix} \cos(ang_y) & 0 & \sin(ang_y) \\ 0 & 1 & 0 \\ -\sin(ang_y) & 0 & \cos(ang_y) \end{bmatrix} \begin{bmatrix} \cos(ang_z) & -\sin(ang_z) & 0 \\ \sin(ang_z) & \cos(ang_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} \cos(ang_y)\cos(ang_z) & -\cos(ang_y)\sin(ang_z) & \sin(ang_y) \\ \sin(ang_z) & \cos(ang_z) & 0 \\ -\sin(ang_y)\cos(ang_z) & \sin(ang_y)\sin(ang_z) & \cos(ang_y) \end{bmatrix} \quad (13)$$

It should be noted that a two-axis rotation is required rather than a three-axis rotation. This is because the aim of the rotation is to compute the sum of the boundary dip angles of all normal vectors. The control variable for the rotation is the current z-axis, and the other normal vectors are performed to follow the same rotation as the current z-axis. The two-axis rotation can uniquely determine the orientation of the current z-axis rotation. Once the normal vectors have been performed by the same rotation of the current z-axis, the sum of the boundary dip angles of all normal vectors can be uniquely determined. Therefore, a two-axis rotation is used instead of a three-axis rotation.

3. Rotate Vec using Rot_i to generate $Vec_i\prime$, then calculate the sum of dip angle sa_i of all boundary normal vectors in $Vec_i\prime$,

4. For each CDP in Vtx, perform step 2 to step 3 to generate the sum of boundary dip angles $Sum_{ang} = \{sa_1, sa_2, \ldots, sa_M\}$ corresponding to all CDPs,

5. Normalize $sa_i \in Sum_{ang}$ as

$$sa_i = \frac{sa_i - \min(Sum_{ang})}{\max(Sum_{ang}) - \min(Sum_{ang})} \quad (14)$$

6. The optimal rotation direction is selected as the CDPs corresponding to $\min(Sum_{ang})$, then rotate Vec accordingly.

To summarize, the main idea of OCM is to find an optimal rotation direction to rotate the current normal vectors so that the sum of the boundary dip angles of all normal vectors after the rotation is minimized. Specifically, given a possible rotation direction dir_p of the current z-axis, let the normal vectors follow the same rotation of the current z-axis, then calculate the sum of the boundary dip angles of normal vectors. This involves two main aspects. First, there are countless dir_p in the whole 3D normal vector space, and the optimal rotation direction dir_p is unknown in advance for an arbitrary set of normal vectors. Therefore, we propose the concept of CDPs to approximate all possible dir_p in the whole 3D normal vector space for selecting the optimal dir_p. Second, the control variable of the rotation in this method is the current z-axis. The rotation of the current z-axis can be uniquely determined by a two-axis rotation, which can also uniquely determine the sum of the boundary dip angles.
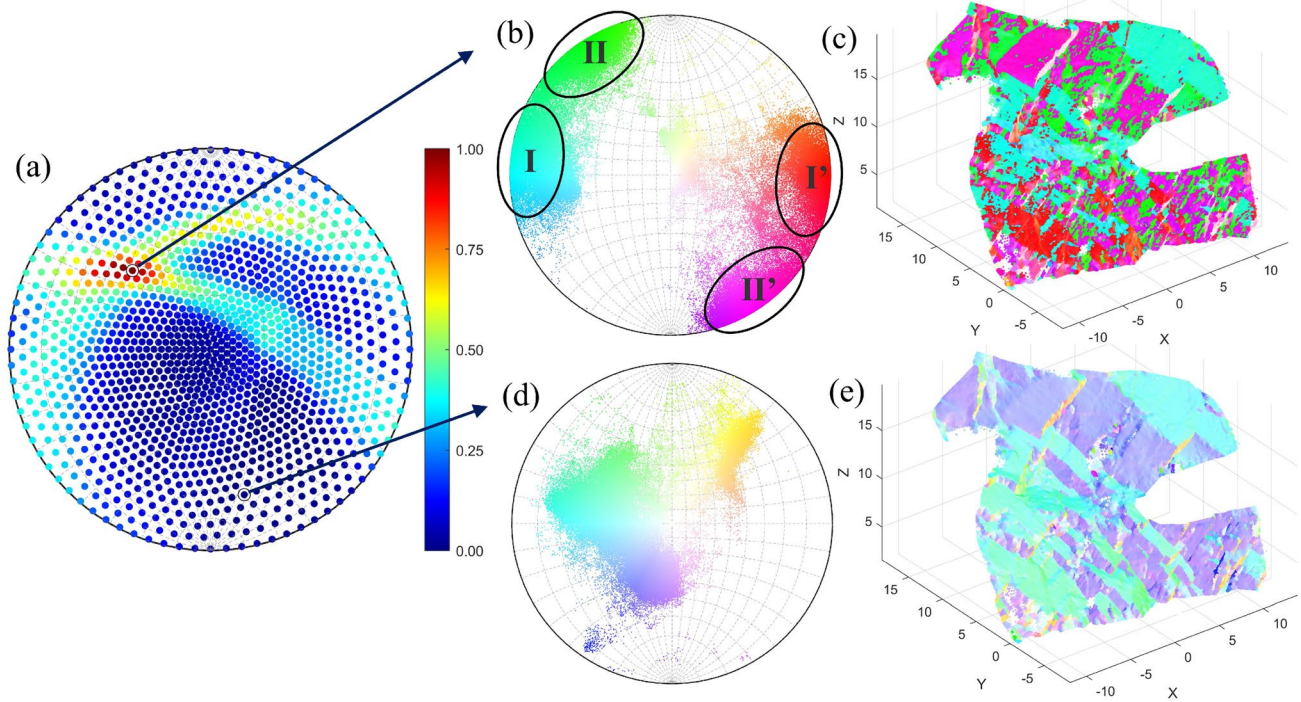
Fig. 8 The optimal and the worst rotation of normal vectors. **a** Sum of dip angles of boundary normal vectors of all CDP. **b** Stereographic projection of the worst rotation. **c** OCM point cloud of the worst rota- tion. **d** Stereographic projection of the optimal rotation. **e** OCM point cloud of the optimal rotation

Therefore, the optimal rotation of normal vectors requires only a two-axis rotation instead of a three-axis rotation.

Figure 8a shows the Sum_{ang} corresponding to all CDPs. Figure 8d, e shows the results of RGB colors mapped with the optimal rotation of normal vectors corresponding to $min(Sum_{ang})$. It can be observed that boundary normal vectors are effectively avoided, and the colors in the same discontinuity plane are uniform and homogeneous, which facilitates the identification of discontinuities by their colors. Comparatively, the results after the worst rotation of normal vectors corresponding to $max(Sum_{ang})$ are shown in Fig. 8b, c. It can be observed that many normal vectors are distrib- uted near the boundaries of the stereographic projection plane, such as the normal vectors in the regions of $I - I\prime$ and $II - II\prime$, resulting in an non-uniform distribution of colors within the same discontinuity plane and making it difficult to distinguish discontinuities by their colors.

The methods in this section are programmed from scratch in Matlab.

2.3 Generation of OCM images (step 3)

After the OCM of normal vectors, the corresponding OCM point cloud can be obtained. This section generates OCM images of OCM point clouds to facilitate the recognition of Mask R-CNN. Considering the direction and density of point

clouds vary with different cases, methods of point cloud direction calibration and image filling are used to generate standard OCM images.

2.3.1 Direction calibration of point clouds (step 3.1)

In this paper, point cloud OCM images are generated from OCM points at the xoz viewpoint. To make discontinuities as perpendicular to the viewing angle as possible, the point cloud model is rotated around the z-axis to make the overall planar fitted vectors parallel to negative y-axis. Given the overall planar fitted vector vec_{mean} calculated by Eq. (1) using all points, vec_{mean} is then projected to the upper unit hemisphere as $vec_{mean} = [x_m, y_m, z_m]^T$. Rotate clockwise around the z-axis to get angle ang_z that makes vec_{mean} negatively parallel with the y-axis. Then ang_z is defined as

$$ang_z = \begin{cases} 180° + \arccos\left(\frac{y_m}{\sqrt{x_m^2 + y_m^2}}\right)° , & x_m > 0 \\ 180° - \arccos\left(\frac{y_m}{\sqrt{x_m^2 + y_m^2}}\right)° , & x_m \leq 0 \end{cases} \tag{15}$$

Equation (15) is programmed using the *acosd* function in Matlab. The point cloud P is rotated clockwise around the z-axis according to ang_z to obtain $P\prime$, and the point cloud
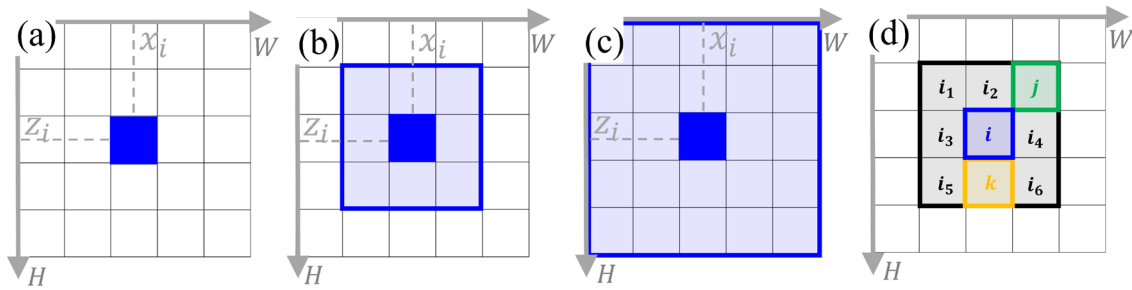
Fig. 9 Illustration of filling length (FL) and void pixels. **a–c** Rectangle filling region with FL of 1, 3 and 5. Blue rectangle regions denote filling regions of a pixel. **d** Void pixel illustration. Black pixel $i_1 \sim i_6$ represent void pixels around pixel i (color figure online)
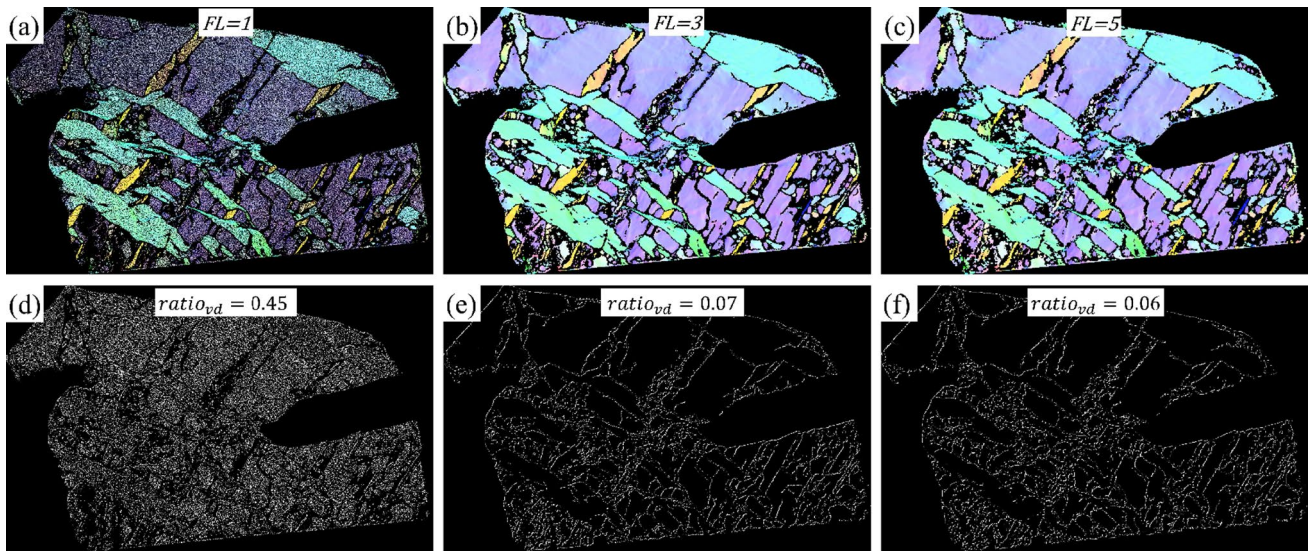

Fig. 10 OCM image generation with different FL. **a–c** OCM images filling with FL of 1, 3, and 5. **d–f** Void pixels corresponding to FL of 1, 3, and 5

OCM image is generated based on the x and z coordinates of all the points in $P\prime$ and their corresponding colors (Fig. 10a).

It should be noted that in the recognition stage of the proposed method, the point cloud orientation calibration needs to be run automatically (e.g., Fig. 10a). While in the data labeling of the training stage, the direction calibration can be replaced by manual point cloud rotations to get the visually most convenient viewpoint for labeling (e.g., Fig. 11i).

2.3.2 Image size calibration and image filling of OCM images (step 3.2)

Because the point cloud varies in size, it first needs to be mapped to a standard OCM image size to facilitate training and recognition. In addition, considering the point cloud is often sparse with different intervals, mapping each point as only one pixel will result in voids in the image (Fig. 10a),

leading to discontinuous colors on the same discontinuity plane. Therefore, this section performs image size calibration and image filling for point clouds.

Given the point cloud coordinate set after Sect. 2.3.1 as $P_{coord} = \{[x_1, x_2, \ldots, x_N]^T, [y_1, y_2, \ldots, y_N]^T, [z_1, z_2, \ldots, z_N]^T\}$ and the corresponding RGB set as $RGB = \{[r_1, r_2, \ldots, r_N]^T, [g_1, g_2, \ldots, g_N]^T, [b_1, b_2, \ldots, b_N]^T\}$. Let the reference OCM image length $L_{img} = 800$, then the calibrated image size is calculated as follows.

First, normalize the coordinates $[x_i, y_i, z_i]^T \in P_{coord}$ of point p_i as

$$
\begin{cases}
x'_i = x_i - \min(x) \\
y'_i = y_i - \min(y) \\
x_i = round\left(\frac{x'_i}{\max[\max(x'), \max(y')]} \times L_{img}\right) + 1 \\
y_i = round\left(\frac{y'_i}{\max[\max(x'), \max(y')]} \times L_{img}\right) + 1
\end{cases}
\tag{16}
$$
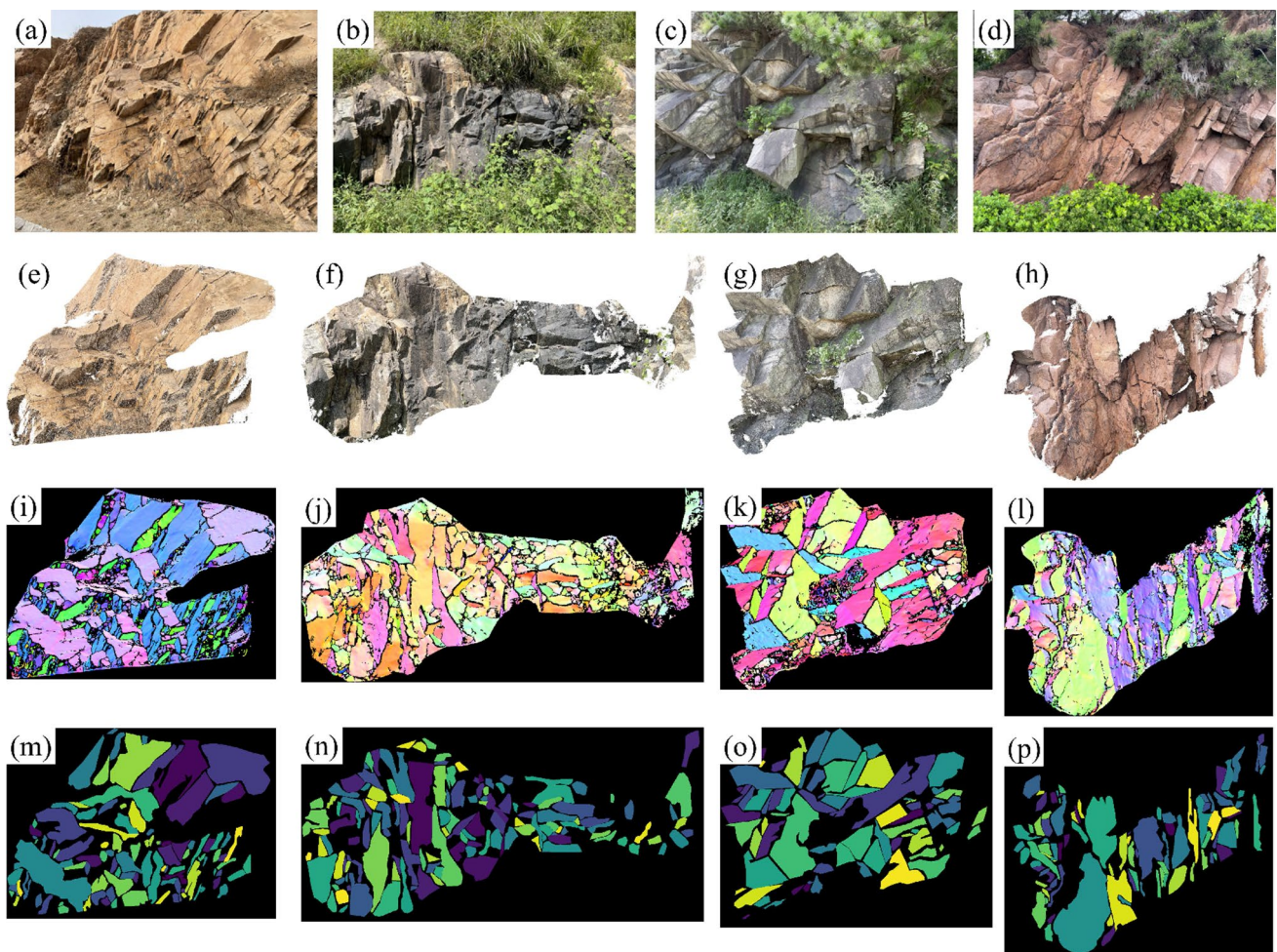
Fig. 11 Data collection and processing examples of the proposed method. **a–d** Images of rock slopes. **e–h** ROI of 3D reconstructed point clouds. **i–l** OCM images obtained using the methods from Sects. 2.1 to 2.3. **m–p** Discontinuity ground truth by manual labeling using software Labelme

where *round* indicates rounding to the nearest integer and is performed using the *round* function in Matlab. Then the image size is set as $H = max(y)$ and $W = max(x)$.

Next, perform the initial generation of OCM images based on the pixel filling. Given a zero image matrix *Img* with the shape of $[H, W, 3]$, the corresponding RGB value of $[x_i, y_i, z_i]^T$ is $[r_i, g_i, b_i]^T$. Then round $[x_i, y_i, z_i]^T$ and fill $[r_i, g_i, b_i]^T$ to the rectangle pixel region with the filling length *FL* and the center location of height z_i and width x_i in *Img* (Fig. 9). During the filling process, set $[r_i, g_i, b_i]^T$ as $[0, 0, 0]^T$ if point p_i belongs to the sharp point set Set_{shp} generated in Sect. 2.1.2. After all the points in P_{coord} are performed by filling, the OCM image can be generated.

There are two reasons for masking non-ROI regions of the OCM image with black. First, it can reduce the interference of non-ROI regions on the discontinuity recognition in ROI regions. Second, the RGB mapping in Sect. 2.2.1 keeps assigning non-black colors to the point cloud by setting $V = 1$, avoiding the point cloud having the same color as the black background of the OCM image, which further reduces the disturbance of non-ROI regions during discontinuity recognition.

A small *FL* can make the intervals in OCM images affect registration (Fig. 8a). However, a large *FL* can cause the excessive overlap of pixels and reduce generation efficiency. Therefore, a void ratio is defined to measure the interval extent of OCM images.

The concept of void pixels is defined as black pixels located at the 8 neighbor pixels of a non-black pixel. For example, as shown in Fig. 9d, the void pixel near the blue i pixel is the black pixel of $i_1 \sim i_6$. The white pixel locations in Fig. 10d–f denote the void pixels corresponding to Fig. 10a–c, respectively. The void ratio is defined to evaluate the filling extent of non-black pixels. Given N_{vd} the number of void pixels and N_{nb} the number of non-black pixels, then the void ratio is defined as

$$ratio_{vd} = \frac{N_{vd}}{N_{nb}} \tag{17}$$

Figure 10a–c shows the OCM images with different void ratios. It shows that the void ratio of 0.45 can cause large numbers of void pixels, resulting in the discrete distribution of color pixels in discontinuities. It also shows that $ratio_{vd} = 0.07$ of $FL = 3$ can generate OCM images with continuous color distribution in discontinuities. However, the $ratio_{vd} = 0.06$ of $FL = 5$ is almost the same as $ratio_{vd} = 0.07$ of $FL = 3$, indicating $FL = 5$ causes more overlapping of pixels and redundant filling calculation expense. Therefore, the default $ratio_{vd}$ is set as 0.1. The point cloud filling requires to be performed with the FL sequence of $[1, 3, 5, \dots]^T$ until $ratio_{vd} \leq 0.1$.

2.4 Data collection and processing (step 4.1)

2.4.1 Dataset description (step 4.1.1)

The dataset of this paper includes 43 3D point cloud models of rock slopes. Forty-two models are rock slope data (Fig. 11a–d) collected from Yangkou ring road in Mountain Lao, Qingdao, China. In the acquisition process, 4–8 images were first taken at different angles in front of the rock mass using an Iphone12 mobile phone with the resolution of 4032×3024. Then the image sequence was sent into the Meshroom (Griwodz et al. 2021) opensource 3D reconstruction software (https://github.com/alicevision/Meshroom) to reconstruct the 3D point clouds. Specifically, after dragging the image sequence into Meshroom, click the start button on the top of the software interface to carry out a fully automatic 3D reconstruction, then export the XYZ and RGB information of the point cloud using the default resolution (The specific operations can refer to Meshroom's tutorial at https://sketchfab.com/blogs/community/tutorial-meshroom-for-beginners). The txt format of point clouds is used as the default, and the reference densities of point cloud cases are described in Sect. 3. In this paper, the only input data to Meshroom is the image sequence, and other parameters such as camera internal and external parameters are automatically calculated and matched by Meshroom using the built-in camera parameter database. Finally, after obtaining the point cloud model, ROI of the point cloud was manually cropped or split into different point cloud models in the CloudCompare software. Figure 11e–h shows the ROI point cloud of Fig. 11a–d. In addition, a publicly available benchmark point cloud model was also adopted for analyzation. This rock slope was located in Ouray, Colorado, US and was scanned by Lato et al (2013) using a laser scanner. The raw data include 1,515,722 points.

The 43 rock slopes were divided into a training set, a validation set, and a testing set with the ratio of 70%, 20%, 10%,

Table 2 Dataset description for training, validation and testing

Dataset	Case number	Date type	Data object
Training set	30	1 point cloud and 29 image sequences	Rock slope
Validation set	9	9 image sequences	Rock slope
Testing set	4	4 image sequences	Rock slope

respectively. Table 2 shows the specific information of each dataset. For each point cloud model, the method of Sects. 2.1–2.3 was used to generate point cloud OCM images. A total of 4,415 discontinuity planes were labeled for all OCM images. Through data enhancement, a total of 4,632 valid point cloud OCM images were obtained, including a total of 430,613 discontinuity planes.

2.4.2 Discontinuity labeling based on OCM images (step 4.1.2)

The Labelme software (Wada 2023) is used to manually and interactively annotate the 43 point cloud OCM images. The process of labeling mainly requires visual judgements to segment regions with similar colors into discontinuity plane polygons. Discontinuity planes are also assigned with different indexes when labeling. In addition, sharp points located near the intersection lines of adjacent discontinuity planes can serve as the auxiliary remind of labeling. The result of labeling are mask images containing discontinuity polygons with different indexes. Figure 11m–p shows the labeling results of Fig. 11i–l. It should be noted that the color of Fig. 11i is a little different from Fig. 10a–c. This is because Fig. 10a–c is color mapped using the automatic direction calibration method of Sect. 2.3.1. All the OCM images must be performed by the automatic direction calibration in the recognition stage without any manual intervention. But Fig. 11i is color mapped in the training stage, allowing to replace the automatic direction calibration by manual rotation of the point cloud for labeling convenience (Sect. 2.3.1).

2.4.3 Augmentation by transformation of HSV, affine, and flipping for OCM images and mask images (step 4.1.3)

Image augmentation is applied to expand the dataset for overfitting reduction and generalization improvement of Mask R-CNN. In this paper, three data augmentation methods are used, including HSV transformation, affine transformation, and image flipping.

The purpose of HSV transformation is to improve the model's performance to recognize different colors. Because the method in this paper essentially identifies

discontinuity planes by the relative color values instead of the absolute color values between adjacent discontinuity planes in the point cloud OCM images, the HSV transformation can increase the model's perception to the relative color values and reduce the overfitting to the absolute color values. Therefore, the HSV transformation used in this paper refers to transforming the Hue values with the S and V values unchanged. Ten H values of $[0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]$ are adopted for the HSV transformation of each OCM image. The nine HSV transformations of Fig. 11i are shown in Fig. 12 (the rest one is Fig. 11i itself).

The affine transformation is then performed on all HSV-transformed images. The purpose of affine transformation is to increase the diversity of discontinuity plane morphology. An affine transformation often includes shearing, translation, rotation, and scaling. To control the deformation of discontinuity planes in a relatively reasonable range, the angular ranges of shear and rotation are set to $[-15°, 15°]$ and $[-90°, 90°]$, respectively. The maximum range of translation is set to be half of the side length of images. In this paper, the scaling transformation is not required because all images are uniformly sized before entering the Mask R-CNN model. The total affine transformation of each image is the combination of the above transformations. In addition, the same affine transformation needs to be applied to the pair of point cloud OCM images and mask images.

Image flipping is performed after the HSV and affine transformations. Horizontal flipping and vertical flipping of an OCM image are performed with the probability of 0.5, respectively. Similar to affine, both the OCM image and the corresponding mask image are required to be performed by the same flipping transformation.

Figure 13 shows the affine and flipping transformation results of Fig. 12.

The methods in this section are programmed from scratch in Python.
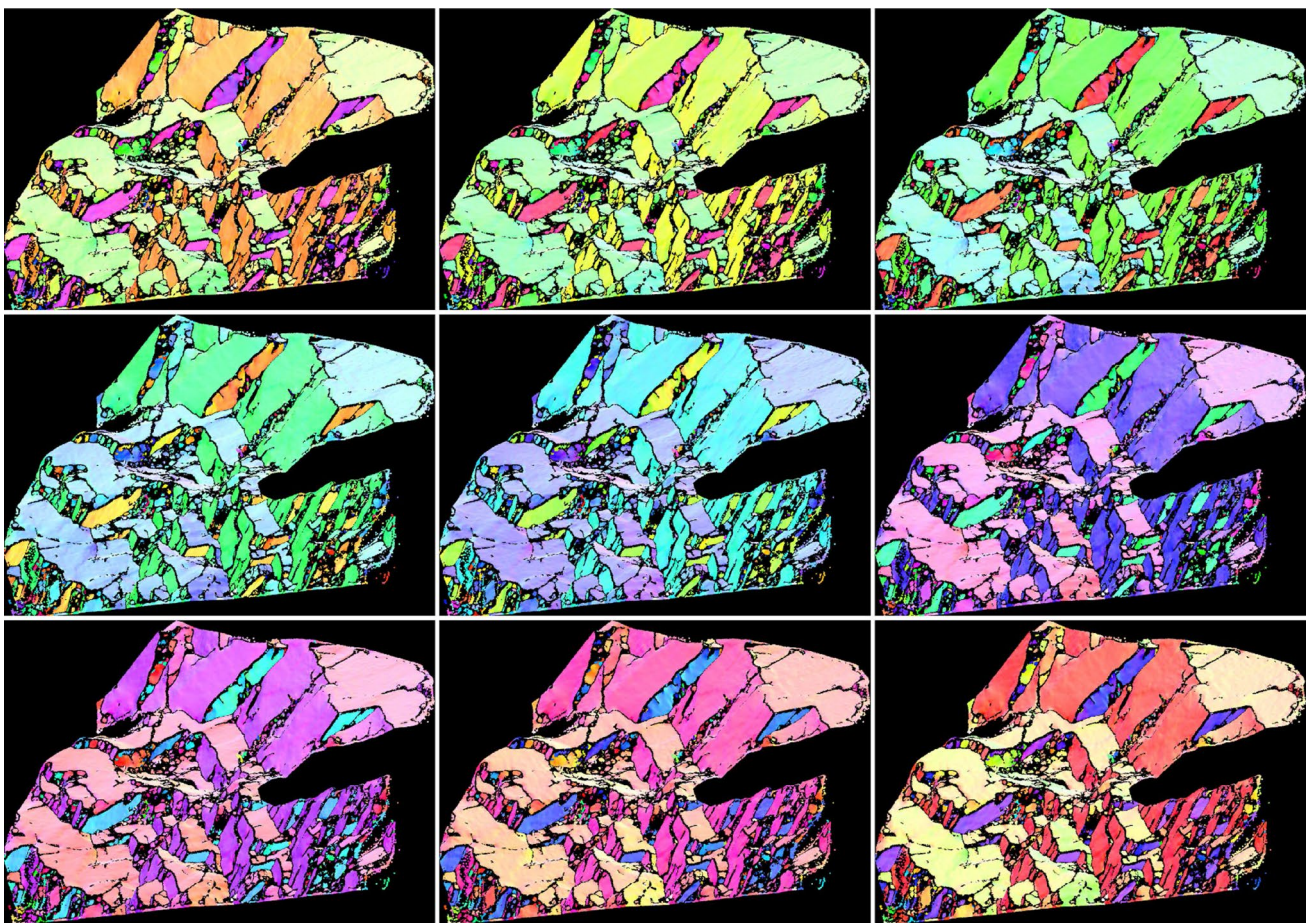
Fig. 12 Data augmentation of HSV transformation. The purpose of HSV transformation is improving the Mask R-CNN's performance to recognize different colors. Because the proposed method essentially identifies discontinuity planes by the relative color values instead of the absolute color values between adjacent discontinuity planes in OCM images, the HSV transformation can increase Mask R-CNN's perception to the relative color values and reduce the overfitting to the absolute color values (Sect. 2.4.3) (color figure online)
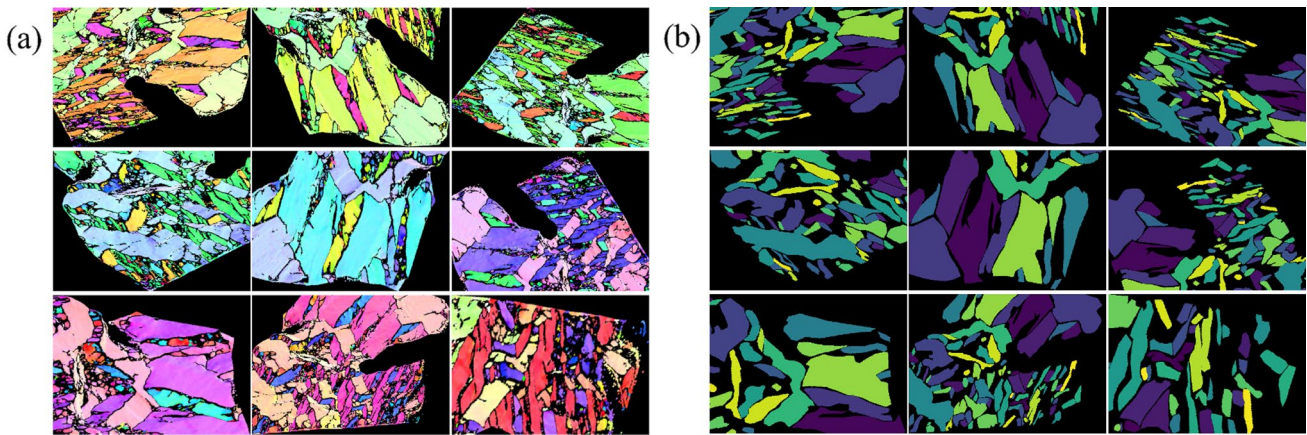
Fig. 13 Data augmentation of affine and flipping transformation. **a** Augmentation of OCM images. **b** Augmentation of ground truth mask images

2.5 Mask R-CNN training (step 4.2)

Mask R-CNN (He et al. 2018) is one of the most classical CNNs for instance segmentation in computer vision field (Agarwal et al. 2019; Gu et al. 2022; Hafiz and Bhat 2020; He et al. 2018). It is typically a two-stage CNN that first generates candidate bounding boxes via a region proposal net (RPN), and then fine-tunes the bounding box while generating pixel-level segmentation within the bounding box, which is well suited for accurately identifying the geometry of discontinuities. In addition, it is simple and flexible to be trained and generalized well in applications (Zaidi et al. 2022). Therefore, Mask R-CNN is adopted for the discontinuity recognition.

2.5.1 Data assignment (step 4.2.1)

As described in Sect. 2.4.1, the augmented dataset in this paper includes a total of 4,632 point cloud OCM images. According to the dataset division in Table 2, OCM images are divided into a training set of 3260 images, a validation set of 1010 images, and a test set of 362 images, which includes 302,425, 94,806, and 33,382 discontinuity planes, respectively. The training set and validation set are involved in the training process. The training set is directly used in the gradient backward propagation, while the validation set is not directly used in training and is only used to generate validation metrics for hyperparameter fine-tuning. The method in this section is programmed from scratch in Python.

2.5.2 Loss function and evaluation metric (step 4.2.2)

2.5.2.1 Loss function According to the initial settings of Mask R-CNN, the loss function of each bounding box of discontinuity planes is set as a multi-task loss as (He et al. 2018)

$$L = L_{cls} + L_{box} + L_{mask} \tag{18}$$

where L_{cls} denotes the binary cross-entropy loss of the bounding box containing discontinuity planes, which is defined as

$$L_{cls} = -[plogq + (1-p)\log(1-q)] \tag{19}$$

where p denotes whether the bounding box contains a discontinuity plane, q denotes the predicted probability that the bounding box contains a discontinuity plane.

In Eq. (18), L_{box} denotes the regression loss of the bounding box.

$$L_{box} = \sum_{i \in \{x,y,w,h\}} smooth_{L1}(t_i - v_i) \tag{20}$$

where (t_x, t_y, t_w, t_h) denotes the predicted values of the bounding box and (v_x, v_y, v_w, v_h) denotes the ground truth of the bounding box; $smooth_{L_1}$ is defined as

$$smooth_{L_1} = \begin{cases} 0.5x^2 & if\,|x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \tag{21}$$

In Eq. (18), L_{mask} denotes the average binary cross-entropy loss of each pixel in the bounding box, indicating

whether a pixel belongs to a discontinuity plane or not. L_{mask} is defined as

$$L_{mask} = -\frac{1}{N_{pix}} \sum_{i=1}^{N_{pix}} \left[p_i log q_i + (1-p_i)\log(1-q_i) \right] \quad (22)$$

where N_{pix} denotes the number of pixels in the bounding box, p_i denotes the ground truth that whether the ith pixel is a discontinuity pixel, and q_i denotes the predicted probability of the ith pixel belonging to a discontinuity plane. The loss functions in this section are programmed using the Pytorch module in Python.

2.5.2.2 Evaluation metric Precision is one of the most effective metrics for measuring model performance in the field of object detection and semantic segmentation (Papandreou et al. 2017; He et al. 2018; Zou et al. 2023). Thus, precision is used for model performance evaluation in Mask R-CNN training (He et al. 2018). Precision is defined as

$$Precision = \frac{TP}{TP + FP} \quad (23)$$

where TP denotes the bounding box containing a discontinuity plane and is predicted as positive, and FP denotes the bounding box does not contain a discontinuity plane but is predicted as positive.

The intersection over union (IOU), one of the important evaluation metrics in the field of image segmentation in computer vision, is used to evaluate the degree of conformity between the predicted mask and the real mask (Ahmed et al. 2015). The higher the IOU, the higher the accuracy of the predicted mask. According to the method of He et al. (2018), this paper adopts the standard COCO (Lin et al. 2015) metrics including AP (average precision, averaged over IOU thresholds of 0.5 : 0.05 : 0.95), AP50 (average precision over the IOU threshold of 0.5), and AP75 (average precision over the IOU threshold of 0.75) to evaluate the model performance, where AP is evaluated using the mask IOU instead of the bounding box IOU.

The methods in this section are performed using the pycocotools-windows module in Python.

2.5.3 Training parameter selection (step 4.2.3)

During training process, an ROI is considered positive if it has IOU with a ground truth box of at least 0.5 and negative otherwise. The mask loss L_{mask} is defined only on positive ROIs. The mask target is the intersection between an ROI and its associated ground truth masks. Each OCM image is set to generate 512 bounding boxes by default, with the ratio between positive and negative samples as 1:1. The non-maximum suppression (NMS) threshold of bounding boxes
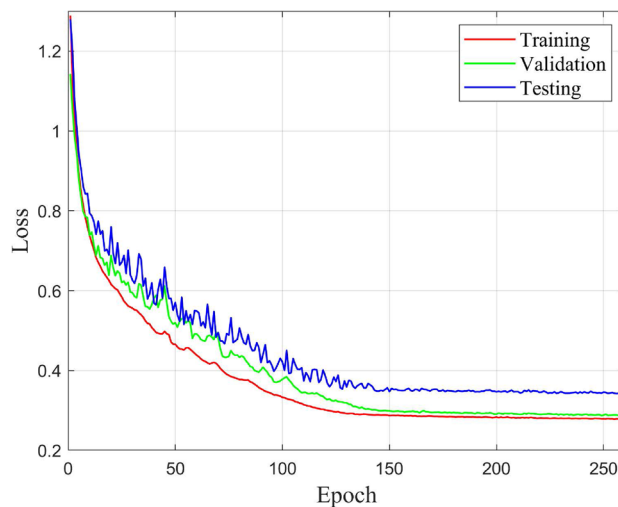
Fig. 14 Loss curvatures of training, validation and testing

is set as 0.7. The minimum probability score of bounding boxes is set as 0.05.

All datasets are trained for a total of 260 epochs (211,900 iterations). The batch size is set to 4 and learning rate is 1e-5. The cosine scheduler is used for the first 150 epochs to discount the learning rate with a decay rate of 0.01. The learning rate is kept unchanged when the epoch number is larger than 150.

The training (and the inferring) process is performed using the Pytorch module in Python.

2.5.4 Training results

Figure 14 shows the loss curvatures of training, validation, and testing. It can be seen that all the training loss, validation loss, and testing loss can effectively reduce within 150 epochs. There is an obvious convergence stage with the epochs larger than 150. Finally, the validation loss is a little bigger than the training loss, and the testing loss is bigger than the validation loss.

Table 3 shows the AP results of the validation set and testing set. It can be seen that AP is 0.616, AP50 is 0.851, and AP75 is 0.725 with the testing set, demonstrating the effectiveness of Mask R-CNN for discontinuity detection and segmentation using OCM images. Table 3 also shows that $AP_{small} < AP_{medium} < AP_{large}$, indicating that Mask R-CNN is better at recognizing large areas of discontinuities in OCM images than small areas of discontinuities.

Figure 15a is the case in the validation set. Figure 15c shows Mask R-CNN's discontinuity recognition results (mask image) of Fig. 15a. It can be seen that the number and shape of the recognized discontinuities are very close to the ground truth (Fig. 15b). Although there is an absence of some small discontinuity planes (areas in the red circles

Table 3 Average precision of validation datasets

Metric	AP	AP_{50}	AP_{75}	AP_{small}	AP_{medium}	AP_{large}
Validation	0.714	0.980	0.894	0.640	0.738	0.872
Testing	0.616	0.851	0.725	0.523	0.609	0.714

*AP, average precision over IOU thresholds of 0.5 : 0.05 : 0.95; AP_{50}, average precision over the IOU threshold of 0.5; AP_{75}, average precision over the IOU threshold of 0.75; AP_{small}, average precision for objects with $area < 32^2$; AP_{medium}, average precision for objects with $32^2 < area < 96^2$; AP_{large}, average precision for objects with area $> 96^2$
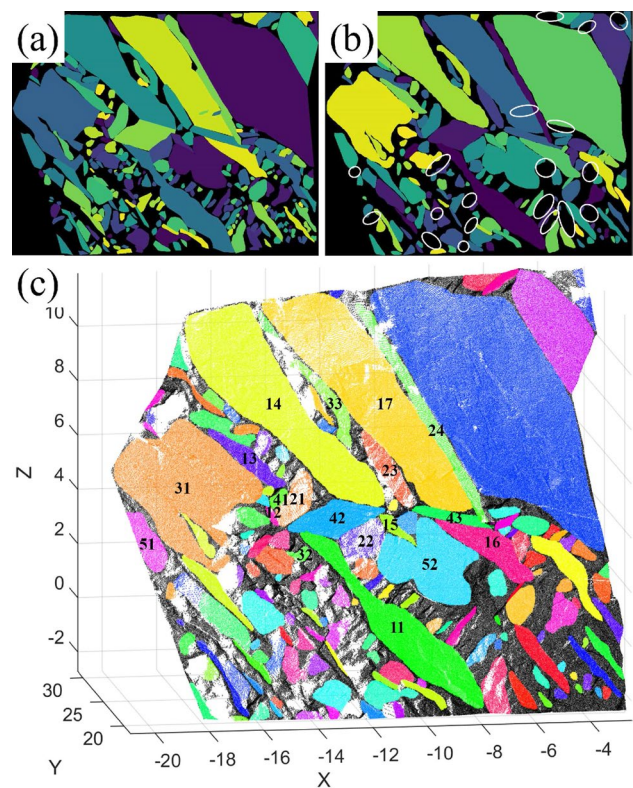

Fig. 15 Discontinuity recognition procedures of the proposed method. **a** OCM image. **b** Discontinuity ground truth by manual labeling. **c** Discontinuity recognition results. **d** 3D discontinuity mapping results
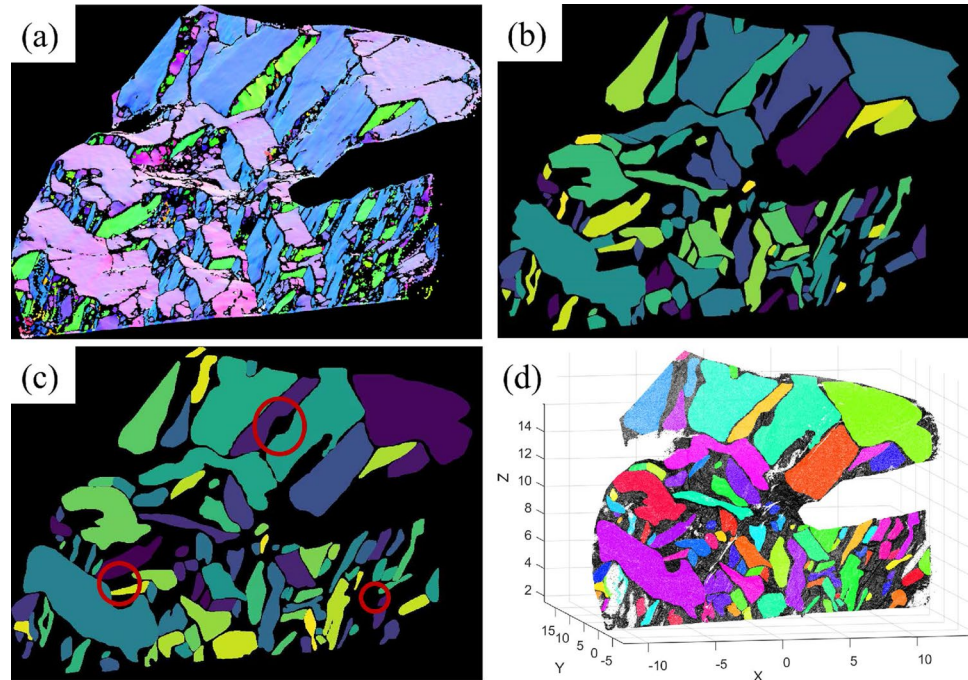
of Fig. 15c), most of the large areas of discontinuities have been recognized correctly.

2.6 3D discontinuity mapping and orientation generation (step 5.2)

The discontinuity recognition results of Mask R-CNN are 2D mask images. Thus, it is necessary to map from the 2D discontinuity mask image to discontinuities in 3D point clouds. Given $P_{coord} = \{ [x_1, x_2, \ldots, x_N]^T, [y_1, y_2, \ldots, y_N]^T, [z_1, z_2, \ldots, z_N]^T \}$ as the point cloud coordinates after the direction calibration in Sect. 2.3.1, Img as the OCM image generated in Sect. 2.3.2, Img_{mask} as the discontinuity mask image recognized by Mask R-CNN and $Label = [l_1, l_2, \ldots, l_N]^T$ as the 3D discontinuity indexes corresponding to P_{coord}.

Since OCM images are generated by x and z coordinates (Sect. 2.3.2), the x and z coordinates in P_{coord} are first rounded to serve as the index of the image coordinates. Considering the size of Img and Img_{mask} are the same, the 3D discontinuity index l_i of each point $[x_i, y_i, z_i]^T \in P_{coord}$ is the pixel value

corresponding at the pixel location of height z_i and width x_i position in Img_{mask}. Figure 15d shows the mapping results of 3D discontinuities of Fig. 15a.

After obtaining the discontinuity indexes of points in the 3D point cloud, all points contained in each discontinuity plane are calculated by Eq. (1) to obtain the discontinuity normal vector vec_p, and then vec_p is projected onto the upper unit hemisphere. Given $vec_p = [x_p, y_p, z_p]^T$, then the dip direction (DD) and dip angle (DA) corresponding to vec_p are calculated as

$$DD_p = \begin{cases} \arccos\left(\dfrac{y_p}{\sqrt{x_p^2 + y_p^2}} \right)^{\circ} & x_p > 0 \\ 360° - \arccos\left(\dfrac{y_p}{\sqrt{x_p^2 + y_p^2}} \right)^{\circ} & \text{otherwise} \end{cases} \quad (24)$$

$$DA_p = \arccos\left(\frac{z_p}{\sqrt{x_p^2 + y_p^2 + z_p^2}} \right)^{\circ} \quad (25)$$

Fig. 16 Data collection and processing of case 1. **a** The rock slope scanned by Lato et al. (2013). ROI is denoted in the black rectangle region. **b** OCM image generated by the proposed method
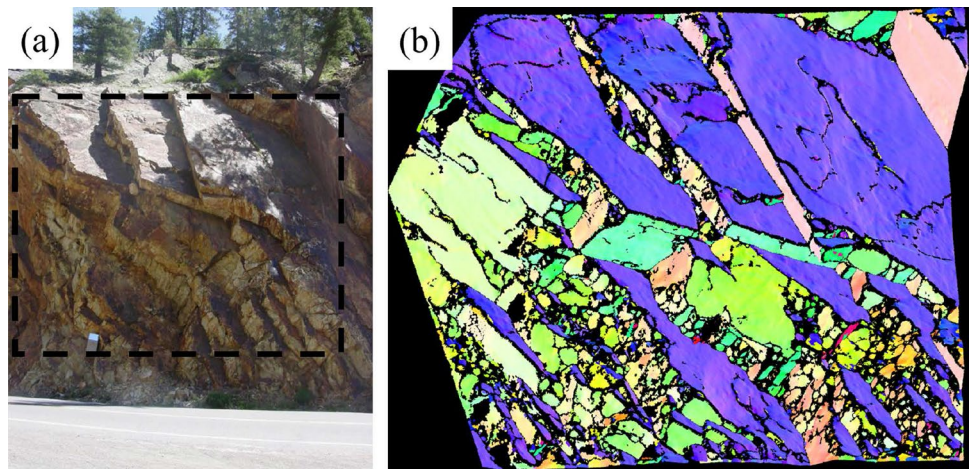
The methods in this section are programmed from scratch in Matlab.

3 Case study

3.1 Case 1: a benchmark rock slope from Lato et al. (2013)

This case is a publicly available point cloud of rock slopes scanned by Lato et al. (2013) (Fig. 16a). The raw point cloud includes 1,515,722 points. After cropping and downsampling, the ROI region (Fig. 16a) contains 414,710 points with the approximate average spacing of adjacent points of 2.41 cm. It is adopted in many studies as a benchmark model for orientation identification validation (Riquelme et al. 2014; Kong et al. 2020; Wu et al. 2020; Daghigh et al. 2022). Representatively, Daghigh et al. (2022) manually determine the discontinuity orientations using the Segment tool in the CloudCompare software. The orientation results are used as the ground truth of this case for comparison. Chen et al. (2016) proposed a fully automated method of discontinuity recognition and analyzed this case. The raw point cloud was first preprocessed into Delaunay triangular meshes. Then mesh normal vectors were clustered into five sets using an improved K-means algorithm. Finally, discontinuity planes were extracted using shared edge connection of triangular meshes. Therefore, this case is used to compare the accuracy of the proposed method with the above methods on this benchmark case.

Figure 16b shows the OCM image of this case. It can be seen that the color within discontinuities is uniform, which effectively avoids the problem of excessive color inconsistence (Figs. 6 and 8b, c). In addition, the sharp points located near intersection lines of discontinuity planes can serve as an effective auxiliary remind for discontinuity segmentation, which is convenient for labeling and recognition. This image is involved

Fig. 17 Discontinuity recognition procedures of case 1 by the proposed method. **a** Discontinuity ground truth by manual labeling. **b** Discontinuity recognition results. **c** 3D discontinuity mapping results

in training as a validation data, and the manually labeled ground truth is shown in Fig. 17a. Figure 17b shows the discontinuity recognition result by Mask R-CNN, which is very close to the manually labeled ground truth (Fig. 17a). Although some trivial discontinuities (areas within the white ovals in Fig. 17b) are missing, major discontinuity planes are effectively identified. The mapping results of 3D discontinuities in point clouds are

Table 4 Orientation comparison of case 1 by different methods

Plane id	Ground truth Daghigh et al. (2022)	Chen et al. (2016)		The proposed method	
		DD/DA	Δ	DD/DA	Δ
11	246.1/39.1	244.6/38.4	1.2	245.8/38.9	0.3
12	248.1/49.5	256.2/52.2	6.8	254.6/51.1	5.2
13	249.8/35.7	251/36.2	0.9	251.7/36	1.2
14	251.8/34.7	251.4/33.9	0.8	252.6/35.6	1
15	249.3/35.4	250.8/36.8	1.7	251.1/35.6	1.1
16	250.1/35.7	250.5/35.9	0.3	250.2/35.7	0.1
17	253.4/33.3	253.2/33.5	0.2	252.1/34.6	1.5
21	339.5/83.1	157.6/83.8	2	338/83.4	1.5
22	345.7/73.2	166.3/78.7	5.5	346.5/74.2	1.3
23	338.2/88.1	157.5/86.9	1.4	159.6/89.7	2.1
24	173.7/76.7	353.1/77.8	1.2	176.5/74.9	3.3
31	136.7/77	314.7/80	3.6	135.4/81.4	4.6
32	131/84	136.5/89.9	8.1	130.8/81	3
33	147.2/89.1	145.6/89.9	1.8	331.4/89.1	4.2
41	97.2/62.2	286/59.8	8.1	100.4/60.1	3.5
42	92.5/48.6	272.6/47.6	1	93.2/48.3	0.6
43	97.7/48.8	277.3/49.3	0.6	97.3/49.6	0.9
51	304.4/77.9	305/77.6	0.7	302.6/77.9	1.8
52	286.7/70.7	109.3/76.6	6.4	286.5/71.3	0.6
Mean error			2.8		1.9
Max error			8.1		5.2

*Δ denotes the angle between planes (°)
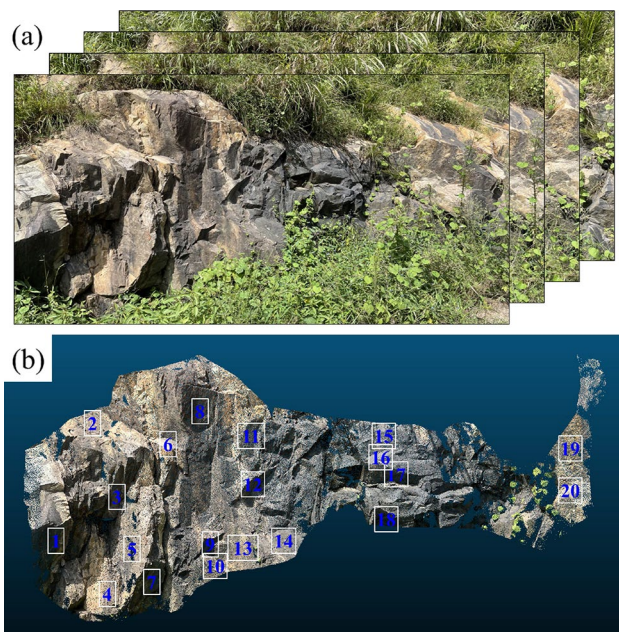
Fig. 18 Data collection and processing of case 2. **a** Image sequence. **b** 3D reconstructed point clouds. Blue numbers denote indexes and locations of manually selected discontinuity planes (color figure online)

shown in Fig. 17c. It can be observed that each discontinuity plane is relatively complete and flat in the 3D point cloud model.

The orientation of each discontinuity plane in Fig. 17c is calculated according to Eqs. (24–25). Table 4 shows the orientation comparison of the proposed method with other methods. It can be seen that the proposed method has the smallest average error of 1.9°, and the maximum error of 5.2° has been effectively reduced compared with the other methods of 8.1°.

This case indicates the effectiveness and accuracy of the proposed method on benchmark rock slope models.

3.2 Case 2: a rock slope

This case is collected from a rock slope of Yangkou ring road in Mountain Lao, Qingdao, China. An Iphone12 mobile phone was used to take four images with the resolution of 4032 × 3024 at different angles in front of the slope (Fig. 18a). The image sequence was processed by the Meshroom software to reconstruct the raw point cloud of 341,611 points. After the cropping without downsampling, the ROI region contained 297,823 points with the approximate average spacing of adjacent points of 2.17 cm. The virtual compass tool in the CloudCompare software was used to interactively select and measure the discontinuity orientation as the ground truth. Figure 18b shows the 20 representative discontinuity planes by manual selection. The corresponding orientations are listed in Table 5. This case is used to compare the accuracy of the proposed method with the fully automated discontinuity identification method of Chen et al. (2016).

Figure 19a shows the point cloud OCM image generated according to Sects. 2.1–2.3, where it can be seen that each discontinuity plane is filled with a uniform color. The sharp points at the intersection lines of adjacent discontinuity planes can serve as an effective remind of segmentation. This case is also used in the validation set. It can be seen from Fig. 19c that the recognized discontinuity planes are very similar to the shapes and locations of the manually labeled discontinuities (Fig. 19b). Figure 19d shows the recognized 3D discontinuities after mapping.

Figure 19e shows the identification results of the method of Chen et al. (2016). An improved K-means method was first used to cluster normal vectors into k from 2 to 6 groups. The Silhouette index was then calculated for each grouping result to select the optimal group number as 3 (Fig. 20). In addition to $k=3$, the grouping results corresponding to $k=4$ and $k=5$ (Fig. 21a, c) with relatively large Silhouette values and the corresponding discontinuity identification results (Fig. 21b, d) were also calculated.

Table 5 shows the orientation error of different methods. The proposed method has the highest accuracy with an average error of only 2.9° and a maximum error of

Table 5 Orientation comparison of case 2 by different methods

Plane id	Ground truth	Chen et al. (2016)						The proposed method	
		k = 3		k = 4		k = 5			
		DD/DA	Δ	DD/DA	Δ	DD/DA	Δ	DD/DA	Δ
1	249.2/40.2	350.4/71.5	15.3	337.4/60.5	1.6	337.8/59.2	1.4	332.6/59.6	5.8
2	264.2/57	185.7/50.5	10.8	185.1/46.9	13.1	185.5/46.2	13.9	172.1/56.7	3.2
3	264/41.9	350.4/71.5	15.2	337.4/60.5	2.8	344.9/59.6	8.1	341.3/62.2	4.3
4	252.6/36.5	42.1/87	27.9	189.1/88.8	5.4	189.9/89.4	4.9	192.6/86.9	1.6
5	248.7/37	42.1/87	12.8	50.6/84.4	4	50.6/84.5	4	51.5/84.5	3.2
6	254.8/29.9	171/37.8	1.7	171.1/37.4	1.3	171.1/37.4	1.2	170.7/37	0.8
7	249.9/35.9	0.7/71.7	21	345.2/56.8	0.7	345/55.8	0.3	343.7/55.9	1.2
8	338.7/82.4	352/80.7	12.8	348.5/76.7	7.6	350.3/76.2	8.3	344.5/77.2	6.9
9	347.5/79	352/80.7	18.6	340.8/64.9	0.8	340.9/64.9	0.8	340.3/64.9	0.8
10	341/89.5	155.1/52.7	2.8	152.1/51.5	4.4	146.4/57.3	7	154.1/54.9	0.7
11	353.5/76.4	353/83.4	10.5	344.9/76.8	0.2	344.9/76.6	0	345.4/77	0.6
12	314.1/77.2	353/83.4	36.1	345.9/79.5	28.2	346.1/79.5	28.3	323.4/63.6	1.7
13	302.4/75.9	43.2/84.4	1.7	44.7/83.9	0.6	45.4/83.6	1	45.1/83.6	0.7
14	330.2/83	168.7/51.9	5.8	167.5/50.6	5.1	168.4/49.3	3.6	171.8/44.6	1.9
15	286.1/58.9	170/69.4	10.4	358/89.9	18.8	183.2/76.3	5.5	177.3/82.5	11.7
16	274.2/51.1	170/69.4	32.5	165.1/66.2	32.6	180.2/55.2	15.8	192.4/45.3	2.2
17	277.2/46.4	352.4/69.2	13.1	352.7/60.5	4.4	352.7/60.6	4.5	352.7/61.7	5.5
18	305/77.6	347.6/79.6	19.7	347.7/75.6	15.7	348.7/60.2	0.5	349/62.1	2.2
19	290.2/67	235.2/88	19.8	235.3/87.9	19.7	235.3/88	19.8	46.2/72.4	2.4
20	236.8/45.7	196.3/54.3	31.8	219.8/45.4	12.1	220.5/45.3	11.6	237.3/47.2	1.5
Mean error			16.0		8.9		7.0		2.9
Max error			36.1		32.6		28.3		11.7

*Δ denotes the angle between planes (°)

11.7°. In comparison, the method of Chen et al. (2016) has the highest accuracy with an average error of 7° and the smallest maximum error of 28.3°. The better performance of the proposed method mainly attributes to the flatness of the recognized discontinuity planes and the shapes well similar to the manually labeled discontinuities. In contrast, the discontinuity recognition effect of Chen et al. (2016) is first affected by the selection of group numbers. Besides, the segmentation of discontinuity planes is inaccurate, making the shape of discontinuities more different from the manual judgements (Fig. 19b). For example, the largest error of 36° mainly attributes to the No. 12 (Fig. 19e) plane which is not separated sufficiently from the No. 11 plane and contains non-in-plane noise. Similarly, the No.16 (error of 32.5°) plane and the No.20 (error of 31.8°) plane are also failed to be well segmented from other planes and noise. However, when the discontinuity plane is well segmented, such as shown in the No.13 plane in Fig. 19e, the corresponding orientation error can be very small as 1.7°.

This case indicates that the proposed method has better accuracy and robustness than the method of Chen et al. (2016).

3.3 Case 3: a rock tunnel excavation face

This case is an excavation face from a rock tunnel in western China. The tunnel is excavated using the drilling and blasting method. The blasting and construction disturbance joints increase the difficulty of discontinuity identification. During the construction gap after blasting slagging and before the steel arch installation, six images with the resolution of 5760×3240 (Fig. 22a) were taken at different angles in front of the excavation face using an Iphone11promax mobile phone. The Meshroom software was used to reconstruct the 3D point cloud model (Fig. 22b) from the image sequence. The raw point cloud contained 1,429,767 points. After the cropping without downsampling, the ROI region contained 639,955 points with the approximate average spacing of adjacent points of 2.91 cm. The virtual compass tool in the CloudCompare software was used to interactively select and measure 20 representative discontinuities (Fig. 22b) as the ground truth.

This case is adopted as a testing data which does not participate in any Mask R-CNN training process. Since the training and validation sets of the Mask R-CNN network only include rock slope data, this case of a rock tunnel
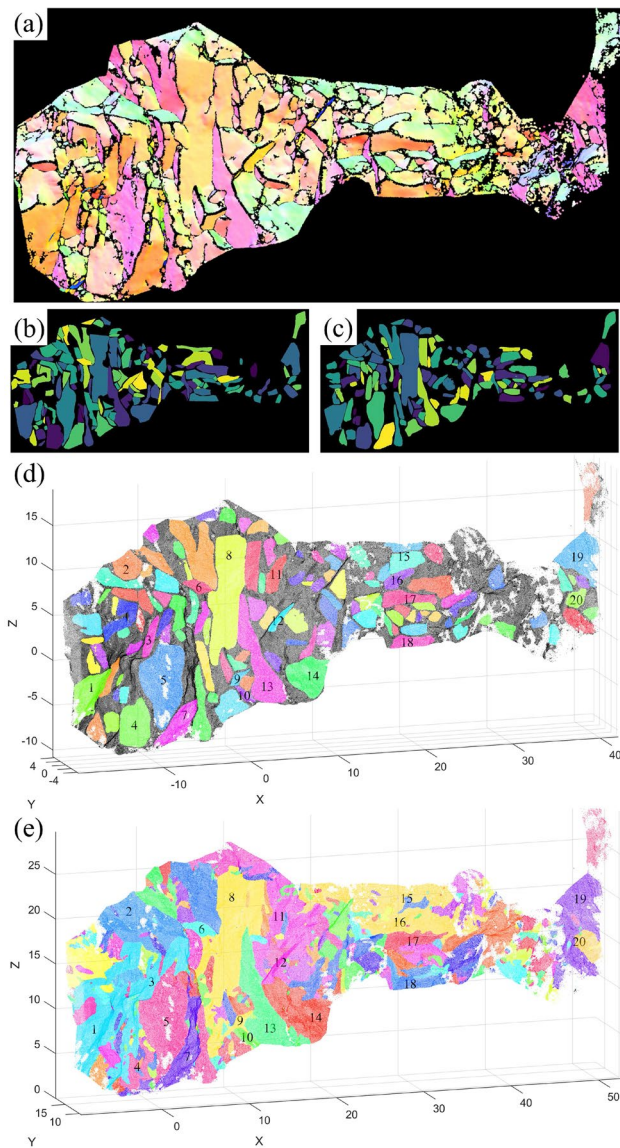
Fig. 19 Discontinuity recognition procedures of case 2 by different methods. **a** OCM image generated by the proposed method. **b** Discontinuity ground truth by manual labeling. **c** Discontinuity recognition results of the proposed method. **d** 3D discontinuity mapping results of the proposed method. **e** Discontinuity recognition results of Chen et al. (2016)

Fig. 20 Silhouette values of case 2 for grouping results of Chen et al. (2016)

excavation face is also used to validate the adaptability and robustness of the proposed method in different scenarios.

Figure 23a–c shows the recognition results of the proposed method. Specifically, Fig. 23a indicates the proposed method can effectively generate point cloud OCM images because the color within each discontinuity plane is uniform. Sharp points are still located at the intersection lines of discontinuity planes, facilitating the recognition and segmentation of discontinuity planes. Figure 23b shows that the discontinuity planes identified by Mask R-CNN can cover most of the main discontinuity planes generated by manual labeling (Fig. 23a). All of the 20

manually selected typical discontinuity planes have been effectively identified. Figure 23c shows the mapping results of 3D discontinuity planes in the point cloud.

Figure 23d shows the discontinuity recognition results of Chen et al. (2016). The optimal grouping results correspond to the largest Silhouette value of $k = 6$ (Fig. 24). It can be seen that there are some flat discontinuity planes similar to the OCM image in Fig. 23a, such as the No. 9 plane in Fig. 23d. However, many discontinuity planes are not segmented effectively, such as the No.3 plane and the No.4 plane in Fig. 23d. Some discontinuity planes (e.g., No.1, 7, 12, 16, 19 planes in Fig. 23d) are not very flat because they contain uneven regions that have non-uniform colors in Fig. 23a. In addition, discontinuity recognition results (Fig. 25b, d) corresponding to other grouping results (Fig. 25a, c) with large Silhouette values of $k = 3$ and $k = 4$ (Fig. 24) were also computed for comparison.

The orientation comparison of different methods is shown in Table 6. It can be seen that the proposed method has the smallest average error (3.1°) and the smallest maximum error (7.8°). Comparatively. The method of Chen et al. (2016) has the smallest average error of 6.2° at the optimal group number of $k = 6$, which are almost twice as much as the proposed method. The maximum orientation error is as high as 35.8° of the No. 1 plane in Fig. 23d. This is because the No. 1 plane is very uneven. Figure 23d shows that the No. 1 plane contains regions with obviously different colors and multiple sharp lines in Fig. 23a. The large orientation error of the No.18 (Fig. 25b) plane and the No.11 (Fig. 25d) plane is also caused by the unevenness of identified discontinuity planes. In contrast, the proposed method performs recognition directly based on the color of OCM images reflecting the flatness, generating flat discontinuity planes that better match the manual labeling results.
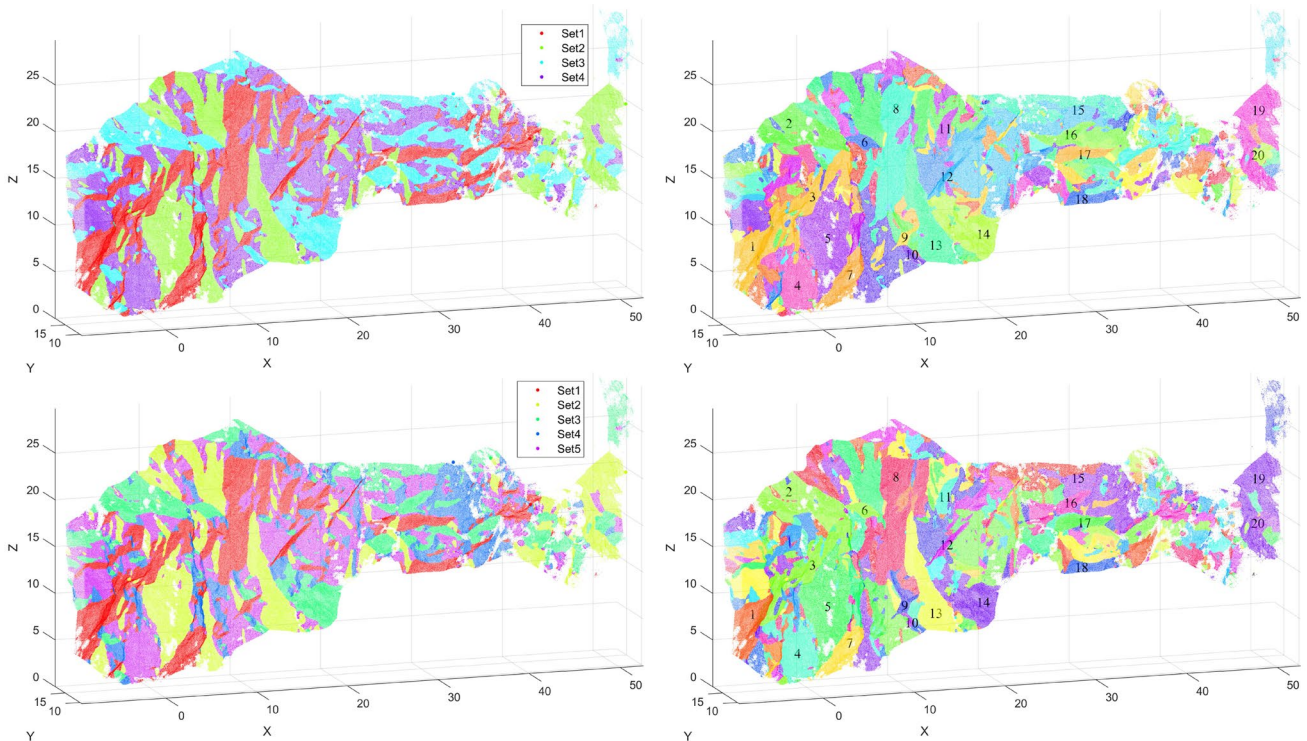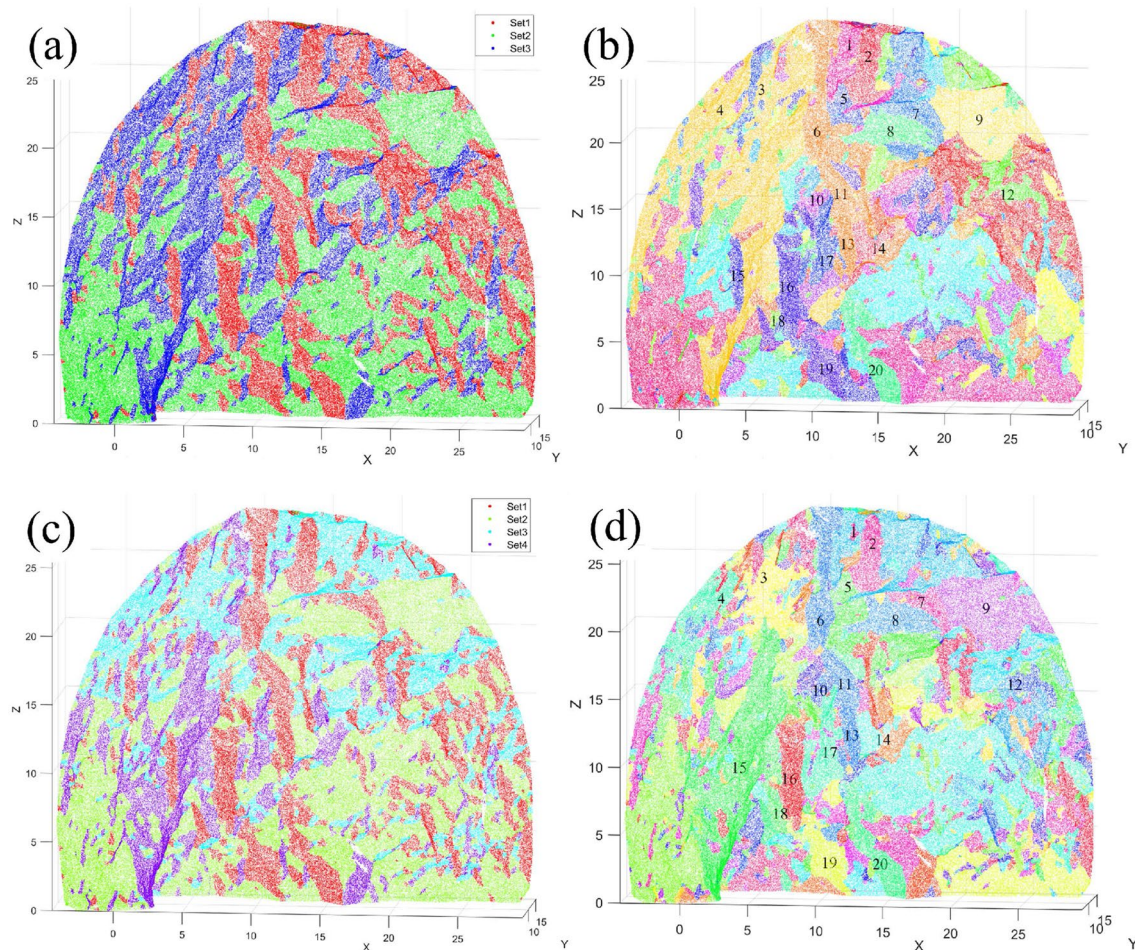
Fig. 21 Orientation grouping and discontinuity recognition results of case 2 by Chen et al. (2016). **a** Orientation grouping results of $k=4$. **b** Discontinuity recognition results of $k=4$. **c** Orientation grouping results of $k=5$. **d** Discontinuity recognition results of $k=5$

Fig. 22 Data collection and processing of case 3. **a** Image sequence. **b** 3D reconstructed point clouds. Blue numbers denote indexes and locations of manually selected discontinuity planes (color figure online)
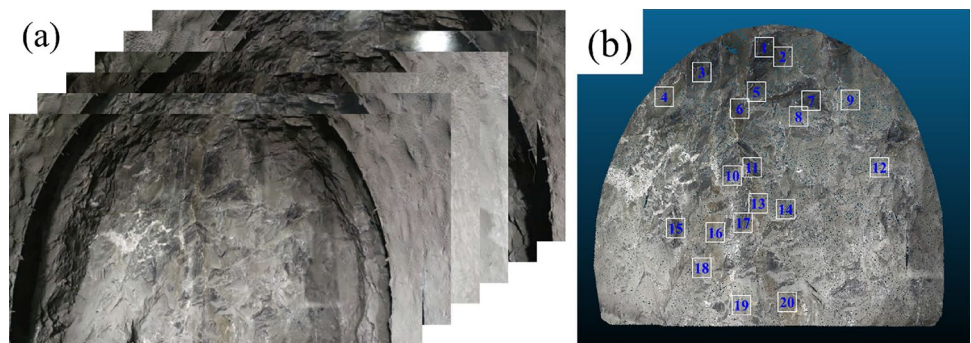
This case illustrates that although only rock slope data are used for the Mask R-CNN training, the proposed method can effectively identify rock tunnel excavation faces, demonstrating the adaptability and robustness of the proposed method for different scenarios.

4 Discussion

4.1 Sensitivity analysis toward point cloud density

The proposed method can identify and generate 3D discontinuity planes without any manual intervention when processing different 3D point cloud models. Therefore, the density of point clouds is critical for the proposed method. To analyze the effect of different point cloud densities on the proposed method, the point cloud of case 2 is randomly resampled using nine downsample ratios. As shown in Table 7, the original case 2 contains 297,823 points. Then nine ratios of 1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, and 0.2 are used to perform the downsampling, generating the downsampled models with 297,823; 268,041; 238,258; 208,476; 178,694; 148,912; 119,129; 89,347; and 59,565 points, respectively. The method of Sect. 2.3.2 is used to generate point cloud OCM images by color filling with the void ratio around 0.1.

Fig. 23 Discontinuity recognition procedures of case 3 by different methods. **a** OCM image generated by the proposed method. **b** Discontinuity recognition results of the proposed method. **c** 3D discontinuity mapping results of the proposed method. **d** Discontinuity recognition results of Chen et al. (2016)
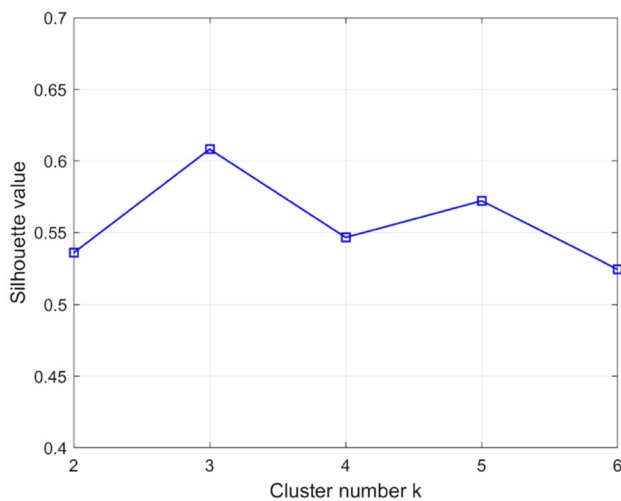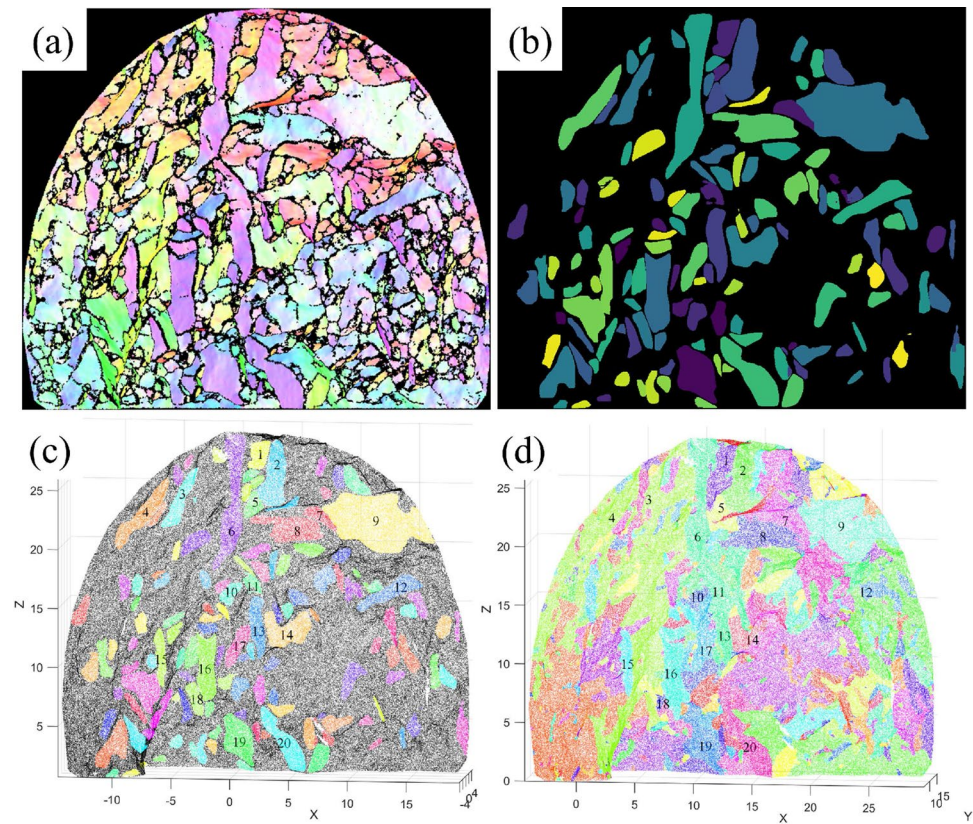
Fig. 24 Silhouette values of case 3 for grouping results of Chen et al. (2016)

The specific parameters of different point cloud density models (M1–M9) are shown in Table 7. Figures 26 and 27 show the OCM images and the recognition results of the proposed method. It can be seen that the void increases and the color in discontinuity planes gradually become discrete as the density decreases. Meanwhile, as shown in Table 7, the number of effectively recognized discontinuities decreases with the downsample ratio increases. However, it should be noted that all of the 20 manually selected representative discontinuity planes (red dot locations in Fig. 27) are effectively identified from M1 to M3. Even when the downsampling changes from M1 to M7, most of the representative discontinuity planes can still be recognized, indicating the recognition effect of main discontinuity planes (i.e., manually labeled representative discontinuity planes) by the proposed method is robust to variations of the point cloud density. However, when the point cloud downsampling rate reaches 0.3 (M8) or even 0.2 (M9), both of the total number of recognized discontinuities and the number of recognized representative discontinuities have been steeply reduced, indicating the overly sparse point cloud can significantly affect the proposed method. Therefore, the point number in 3D point cloud models is suggested to be larger than about 25% of the reference image pixels with the resolution of 800×800, which is 160,000 points.

4.2 Efficiency of the proposed method

The proposed method contains two operation stages of training and recognition after acquiring the raw 3D point cloud. All algorithms are programmed using the combination of Matlab (2022a) and Python. All the

Fig. 25 Orientation grouping and discontinuity recognition results of case 3 by Chen et al. (2016). **a** Orientation grouping results of $k=3$. **b** Discontinuity recognition results of $k=3$. **c** Orientation grouping results of $k=4$. **d** Discontinuity recognition results of $k=4$

programs are performed on a Windows platform of an Intel CPU I7 13700K, GPU NVIDIA 4090 and RAM 64GB. The specific running time for the two stages is shown in Table 8. The training stage starts with manual labeling, which takes about 2 h per OCM image, thus the manual labeling of the 43 original OCM images takes about 86 h in total. Then Mask R-CNN training has run for 260 epochs (211,900 iterations) with a total of about 20 h. The main operation time in the training stage is about 106 h. In the recognition stage, it takes 12 s to process a case on average, including 6 s for the NPW-OC contraction, 1 s for normal vector optimal RGB transformation, 2 s for point cloud OCM image generation, and 3 s for Mask R-CNN-based discontinuity recognition and orientation calculation. The good efficiency of the proposed method mainly attributes to the conversion from the direct recognition of large-scale 3D point clouds to Mask R-CNN's recognition of 2D OCM images with fixed sizes, which effectively reduces the iterative calculation of 3D point clouds with different densities and improves the efficiency stability.

4.3 Analysis of characterization and rationality for the proposed method

Different from the traditional methods of discontinuity recognition that directly process point cloud with orientation data (Riquelme et al. 2014; Chen et al. 2017; Ge et al. 2018; Kong et al. 2020; Singh et al. 2021), the proposed method uses OCM images to reflect both the orientation as well as the spatial information of the point cloud. Combined with deep learning methods, the proposed method switches the direct recognition of 3D point clouds into the implicit recognition of 2D OCM images by Mask R-CNN, aiming to improve the performance in the following three aspects compared with traditional methods:

1. Recognition efficiency.

The recognition efficiency of traditional methods is sensitive to the number of points because 3D point clouds are required to be directly processed. In contrast, the proposed

Table 6 Orientation comparison of case 3 by different methods

Plane id	Ground truth	Chen et al. (2016)						The proposed method	
		$k=3$		$k=4$		$k=6$			
		DD/DA	Δ	DD/DA	Δ	DD/DA	Δ	DD/DA	Δ
1	322.7/54.7	323.9/57.6	3.1	2.9/57.6	33.2	5.8/58	35.8	322.6/53.9	0.8
2	40.8/70.6	28.4/66.3	12.3	34.9/72	5.8	35.1/72.3	5.7	34.7/72.4	6.1
3	343.5/60.7	338.9/83.5	23.2	340.2/67.1	7	343.9/60.8	0.4	341.6/66.5	6.1
4	322.2/56.7	338.9/83.5	30.9	337.3/68.6	17.8	320.5/59.5	3.1	321.7/59	2.4
5	180.9/81.6	164.1/83.5	16.8	164.4/83.8	16.4	171.9/81.8	8.9	175.4/81.8	5.4
6	39.9/79.2	20.3/74.9	6.8	32.9/75.6	7.7	37.9/77.7	2.4	34.2/77.5	5.8
7	28.9/71	22.9/65.4	7.9	24.1/81.5	11.5	24.9/80.4	10.1	29/72.8	1.8
8	188.8/59.1	187/62.3	3.5	187/62.3	3.6	188.7/59.8	0.7	180.9/62.9	7.8
9	175.5/85.9	177.4/85.8	1.9	177.5/85.7	2	176.2/87.7	1.9	178.4/86	2.8
10	216.6/53.5	213.5/55.1	3	212.9/55.7	3.7	216.8/56	2.6	215.2/55.1	1.9
11	22.3/74.8	20.3/74.9	2	32.9/75.6	10.2	30.9/80	9.9	22.4/80.3	5.5
12	205.5/69.6	192.2/79.1	16	192/79.5	16.3	200.8/77.4	9	205/71.7	2.2
13	46.7/85.9	20.3/74.9	10.2	32.9/75.6	8.1	30.9/80	7.9	45.1/84.7	2
14	154.8/85.4	158.9/81.1	5.9	158.9/81.2	5.9	162/85.9	7.2	154.7/84.9	0.5
15	32.3/75.8	34.6/77.3	2.7	35.4/77.5	3.4	35.4/77.5	3.4	34.8/76.8	2.6
16	45.9/88.5	42.2/87	4	43.2/87.5	2.9	43.9/87.9	2	45.9/87.4	1
17	316.6/82.9	320.4/79.7	4.9	324.2/80.6	7.8	313/80.2	4.5	318.9/82.9	2.3
18	133.5/68.4	167.3/71.1	31.8	315.2/88	23.7	132/66.4	2.4	131.8/67.2	2
19	35.2/89.5	216/87.5	3.1	216/87.6	3	216.1/87.9	2.8	216/88	2.6
20	42.2/88.2	216.4/85.1	6.5	218.6/87.1	3.7	218.9/87.4	3.3	42.4/87.4	0.8
Mean error			9.8		9.7		6.2		3.1
Max error			31.8		33.2		35.8		7.8

*Δ denotes the angle between planes (°)

Table 7 Information and recognition results of downsample models of case 2

Model	Downsample ratio	Point number	Point number/Non-black pixel number	FL	ratio$_{vd}$	N_D	N_{RD}
Ground truth	1	297,823	0.47	3	0.07	143	20
M1	1	297,823	0.47	3	0.07	119	20
M2	0.9	268,041	0.42	3	0.091	110	20
M3	0.8	238,258	0.37	3	0.103	102	20
M4	0.7	208,476	0.33	3	0.105	96	18
M5	0.6	178,694	0.28	3	0.107	84	17
M6	0.5	148,912	0.23	5	0.085	91	17
M7	0.4	119,129	0.19	5	0.106	65	18
M8	0.3	89,347	0.14	7	0.093	63	14
M9	0.2	59,565	0.09	9	0.092	63	11

*N_D: recognized discontinuity number; N_{RD}: recognized representative discontinuity number

method maps the point cloud into an OCM image of fixed sizes (800×800, Sect. 2.3.2) for processing, and Mask R-CNN is also efficient in recognizing 2D images (He et al. 2018), which enables a stable and efficient recognition of point clouds with different point numbers.

2. Recognition automation.

Traditional methods often require manual fine-tuning of parameters for recognizing different rock models (Riquelme et al. 2014; Kong et al. 2020). Comparatively, the proposed method can finish all the tedious and time-consuming labeling by manual interactions in the training stage, resulting
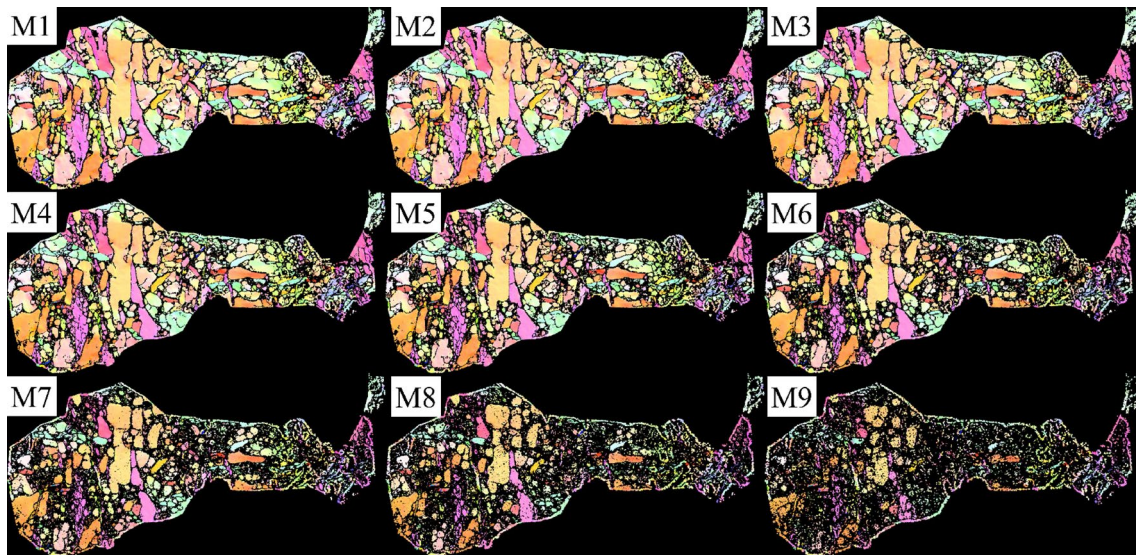
Fig. 26 OCM images of point cloud models with different densities
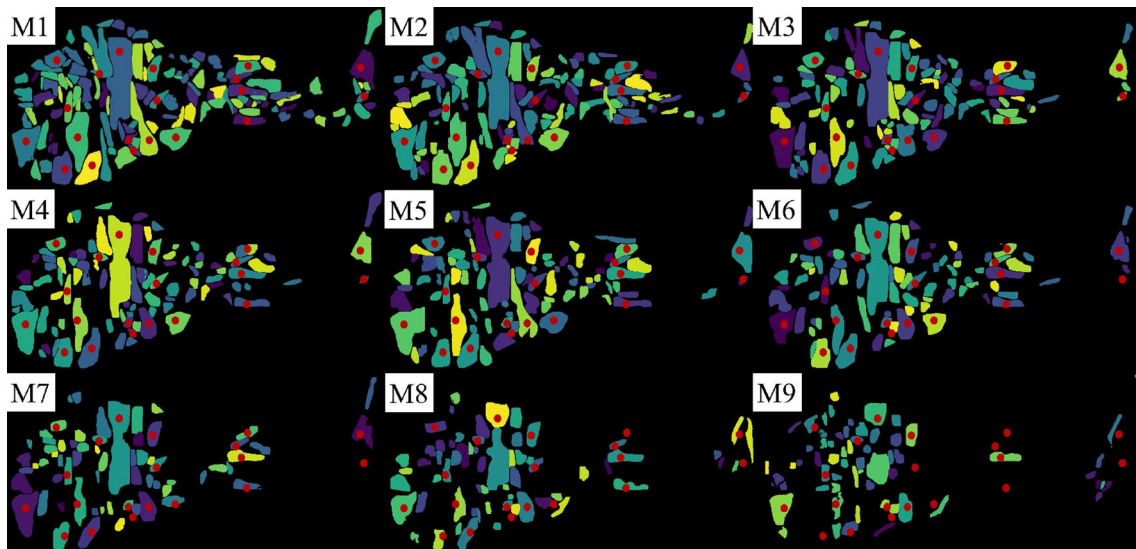

Fig. 27 Discontinuity recognition results of point cloud models with different densities. Red points denote the locations of 20 representative discontinuity planes by manual selection (color figure online)

Table 8 Running time of the proposed method

Operations	Running time	
	Training stage (h)	Recognition stage (s)
OCM image labeling	86h	
Mask R-CNN training	20h	
NPW-OC contraction		6s
Optimal RGB transformation of normal vectors		1s
Generation of point cloud OCM images		2s
Mask R-CNN predicting and orientation generation		3s
Total time	106 h	12s

in the intelligent recognition without manual fine-tuning of parameters for different models during the recognition stage.

3. Proximity of the recognition results to manual judgements.

Traditional methods often need to control the recognition effect using uniform parameter settings of the algorithm (Zhang et al. 2018; Singh et al. 2021), which is ineffective to adjust the morphology of individual discontinuity planes. In contrast, the proposed method can directly edit the morphology of each individual discontinuity plane in the training stage by careful manual annotation, making the morphology of individual discontinuity planes generated by the recognition stage closer to manual judgements.

Besides, the proposed method shows the generalization to different scenarios, which is mainly because the OCM image only depends on the geometrical properties of rock mass and is independent of scenarios and lithologies. As analyzed in Sect. 3.3, the proposed method can be effectively applied to the discontinuity recognition of the rock tunnel excavation face by training with only rock slope data, demonstrating the generalization to different scenarios of rock engineering.

From the applicability point of view, since OCM images are generated from 3D point clouds and normal vectors, the applicability of the proposed method also fundamentally depends on the accuracy and density of point clouds. As analyzed in Sect. 3 and shown in Figs. 16, 18, and 22, effective recognition results can be generated when the approximate average spacing of adjacent points is about 2–3 cm. Both the image-based 3D reconstruction method (case 2 and case 3) and the 3D laser scanning method (case 1) can generate effective point cloud for the recognition of the proposed method. However, it should be noted that when the point cloud is too sparse, the morphology of discontinuity planes in OCM images is incomplete (Fig. 26), which affects with the recognition effects (Fig. 27). In addition, although some point cloud acquisition techniques (such as 3D laser scanning) can collect point cloud with very high resolution (such as the Z + F Imager 5016 laser scanner can reach the resolution of 0.6 mm at 10 m), too dense point cloud is also unnecessary for the proposed method. This is mainly because the size of an OCM image is fixed at 800×800 (Sect. 2.3.2), and too dense point clouds will cause the same pixel of the OCM to be repeatedly colored by different points, resulting in the redundancy of point cloud data. Therefore, different point cloud acquisition techniques can be applied to the proposed method as long as the density of the acquired point cloud is suitable.

In addition, the proposed method also has some limitations. First, considerable manual interaction and advanced knowledge are required in the field during the model training stage. The labeling operation is tedious and time-consuming. Second, the characteristics of rock discontinuities are diverse, making it difficult to recognize complex and random cases only by training a limited amount of data. Third, too sparse point cloud can make the discontinuity planes in OCM images incomplete (Fig. 26), which significantly affects the recognition effect (Fig. 27). Finally, because the neural network is used for an implicit recognition, it is difficult to explicitly adjust and control the recognition effect by manual setting algorithm parameters as traditional methods when the recognition effect is unsatisfactory.

4.4 Applications of the proposed method

The recognized 3D discontinuity planes and orientations can be further used for applications such as rock discontinuity description, geological modeling, rock quality evaluation, and rock numerical analysis (Zhu et al. 2016; Li et al. 2019; Zhang et al. 2020, 2021; Cai et al. 2022). In this section, three applications are taken for example, including orientation grouping, 3D trace length distribution analysis, and discrete fracture network (DFN) generation.

In terms of orientation grouping, given $P_D = \{p_1, p_2, \ldots, p_{DN}\}$ (DN denotes the number of all points belonging to discontinuity planes) the coordinates of all points belonging to discontinuity planes, and $Plane = \{pl_1, pl_2, \ldots, pl_{DM}\}$ (DM denotes the number of discontinuity planes) the index set of discontinuity points, then the normal vectors of points belonging to the ith discontinuity plane pl_i are the same, which can be calculated by Eq. (1). After obtaining the normal vectors of all points in P_D, the improved K-means algorithm of Chen et al. (2016) is used to perform orientation grouping with the group number k set from 2 to 6. The grouping quality is evaluated using the Silhouette index to determine the optimal group number and the corresponding grouping results. The Silhouette value of the ith point in P_D is calculated as

$$S(p_i) = \frac{b(p_i) - a(p_i)}{\max\{a(p_i), b(p_i)\}} \tag{26}$$

where $a(p_i)$ is defined as the average distance of p_i to all other points in the same group, and $b(p_i)$ is defined as the minimum of the average distance between p_i and points in other groups. The final Silhouette value is the mean value of all Silhouette values of points in P_D. A large Silhouette value indicates a good grouping quality.
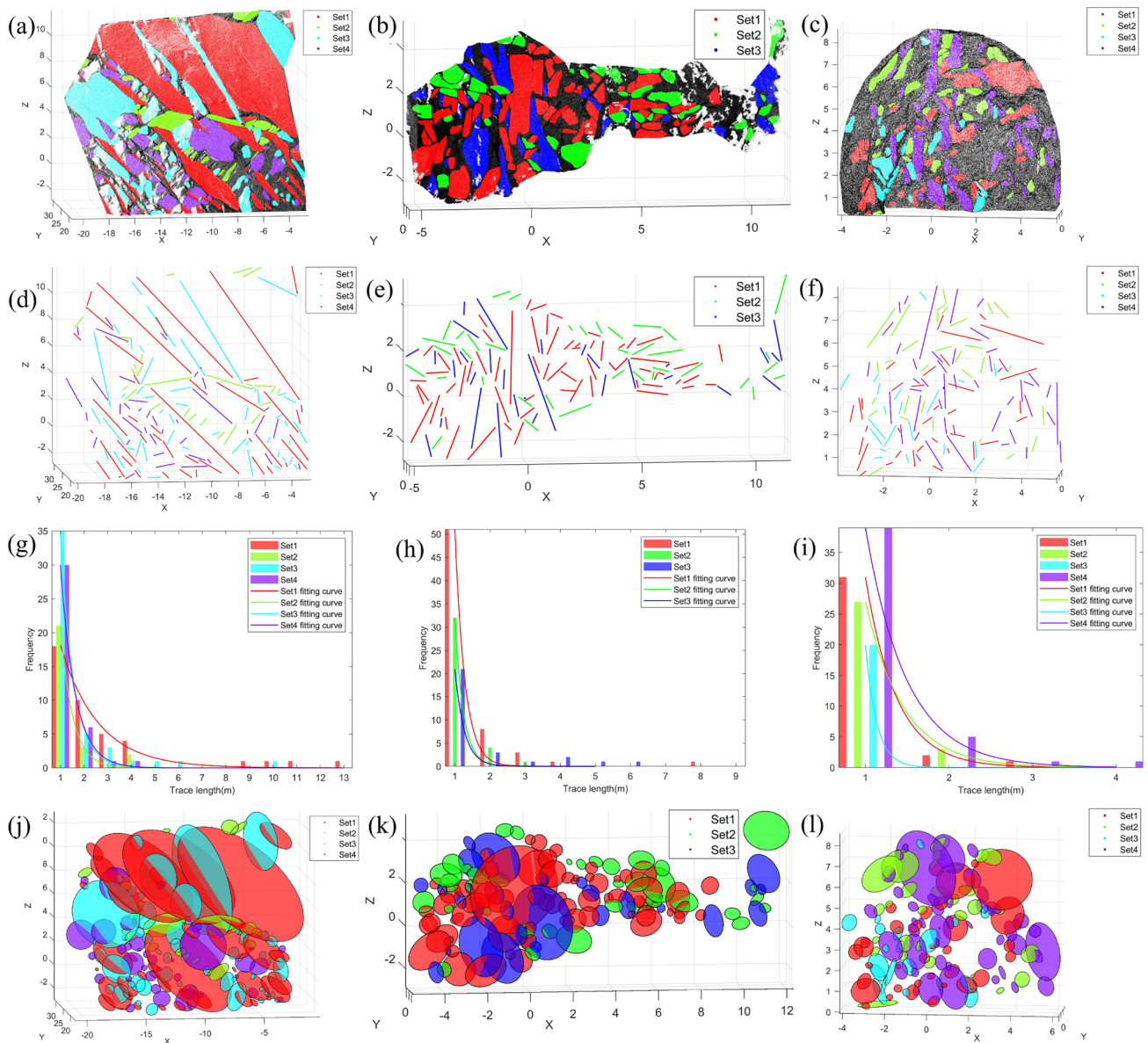
Fig. 28 Applications of the proposed method on case 1–3. **a–c** Orientation grouping results of case 1–3. **d–f** 3D trace recognition results of case 1–3. **g–i** 3D trace length distribution fitting results of case 1–3. **j–l** DFN generation results of case 1–3

Figure 28a–c illustrates the optimal *K*-means grouping results of case 1–3 with the optimal group numbers of $k=3$, $k=4$ and $k=3$, respectively.

As for trace generation and statistical analysis, Laux and Henk (2015) and Riquelme et al. (2018) consider the exposed discontinuity surface as a polygon. They hold that the distance between the two farthest points in the discontinuity point set represents the trace length of the discontinuity. Therefore, given p_{i1} and p_{i2} are the farthest points in discontinuity plane pl_i, then the 3D trace line is

defined as the line from p_{i1} to p_{i2}, and the trace length len_i of pl_i is defined as

$$len_i = ||p_{i1} - p_{i2}|| \tag{27}$$

Figure 28d–f shows the 3D trace results corresponding to Fig. 28a–c, respectively. In addition, the negative exponential function is often used to fit the distribution of trace length (Zhang and Einstein 1998, 2000). The fitting results of trace length distribution are shown in Fig. 28g–i.

DFN is an important basis for rock property analysis (Guo et al. 2022). In DFN, discontinuity planes are often

represented by circular discs to simulate the persistence of fractures in 3D space (Zhang and Einstein 2000). The 3D spatial disc of the discontinuity plane is determined by the center, radius, and normal vector, corresponding to the midpoints of 3D trace lines, the half-length of 3D trace length, and the normal vector of discontinuity planes, respectively. Figure 28j–l shows the DFN models corresponding to case 1–3.

5 Conclusion

This paper proposes an intelligent recognition of rock discontinuity based on OCM of 3D point clouds via deep learning, named OCM. The innovative contributions consist: (1) A neighborhood PCA-weighted oriented contraction (NPW-OC) method is proposed to extract point cloud skeletons as discontinuity intersection lines, (2) OCM of normal vectors based on optimal transformation among normal vectors, HSV and RGB, (3) generation of OCM images based on point cloud direction calibration and filling, (4) OCM image augmentation based on HSV transformation, affine transformation, and image flipping, (5) deep-learning-based discontinuity recognition of Mask R-CNN toward OCM images, and (6) 3D discontinuity mapping from OCM recognition results.

Forty-two rock slope image sequences and a rock slope point cloud are collected and labeled for processing, generating a total of 4,632 OCM images including 430,613 discontinuity planes after data augmentation for training the Mask R-CNN. Two rock slope cases and a rock tunnel excavation face case are adopted for testing. The average recognition time per 3D point cloud model is approximately 12 s, and the total training time is about 106 h. The results show the proposed method can effectively recognize discontinuity planes with shapes, locations, and orientations close to manual recognition results. The proposed method is more accurate than traditional methods and shows good adaptability to different rock engineering scenarios. A sensitivity analysis is performed to validate the robustness of the proposed method to the variations of point cloud densities. In addition, the recognized discontinuity results are applied to three rock engineering practices including orientation grouping, 3D trace length distribution analysis, and DFN generation.

Future research can focus on the efficiency improvement of manual labeling during the training stage, the generalization improvement by increasing the scale and quality of training datasets, the robustness improvement for processing too sparse point cloud, and the exploration of more powerful neural network structures.

Declarations

Conflict of interest No financial or non-financial interest conflicts are directly or indirectly related to the submission of this manuscript.

References

Abellán A, Oppikofer T, Jaboyedoff M, Rosser NJ, Lim M, Lato MJ (2014) Terrestrial laser scanning of rock slope instabilities. Earth Surf Process Landf 39:80–97. https://doi.org/10.1002/esp.3493

Agarwal S, Terrail JOD, Jurie F (2019) Recent advances in object detection in the age of deep convolutional neural networks. https://doi.org/10.48550/arXiv.1809.03193

Ahmed F, Tarlow D, Batra D (2015) Optimizing expected intersection-over-union with candidate-constrained CRFs. In: 2015 IEEE international conference on computer vision (ICCV). Presented at the 2015 IEEE international conference on computer vision (ICCV), pp 1850–1858. https://doi.org/10.1109/ICCV.2015.215

Alférez GH, Vázquez EL, Martínez Ardila AM, Clausen BL (2021) Automatic classification of plutonic rocks with deep learning. Appl Comput Geosci 10:100061. https://doi.org/10.1016/j.acags.2021.100061

Au OK-C, Tai C-L, Chu H-K, Cohen-Or D, Lee T-Y (2008) Skeleton extraction by mesh contraction. ACM Trans Graph 27:1–10. https://doi.org/10.1145/1360612.1360643

Barton N (1978) Suggested methods for the quantitative description of discontinuities in rock masses: International Society for Rock Mechanics. Int J Rock Mech Min Sci Geomech Abstr 15:319–368

Cai W, Zhu H, Liang W (2022) Three-dimensional tunnel face extrusion and reinforcement effects of underground excavations in deep rock masses. Int J Rock Mech Min Sci 150:104999

Caliński T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat 3:1–27. https://doi.org/10.1080/03610927408827101

Cao J, Tagliasacchi A, Olson M, Zhang H, Su Z (2010) Point cloud skeletons via Laplacian based contraction. In: 2010 Shape modeling international conference. Presented at the 2010 shape modeling international (SMI), IEEE, Aix-en-Provence, France, pp 187–197. https://doi.org/10.1109/SMI.2010.25

Cawood A, Bond C, Howell J, Butler R, Totake Y (2017) LiDAR, UAV or compass-clinometer? Accuracy, coverage and the effects on structural models. J Struct Geol 98:67–82. https://doi.org/10.1016/j.jsg.2017.04.004

Chen J, Huang H, Cohn AG, Zhang D, Zhou M (2021) Machine learning-based classification of rock discontinuity trace: SMOTE oversampling integrated with GBT ensemble learning. Int J Min Sci Technol S2095268621000896. https://doi.org/10.1016/j.ijmst.2021.08.004

Chen J, Zhu H, Li X (2016) Automatic extraction of discontinuity orientation from rock mass surface 3D point cloud. Comput Geosci 95:18–31. https://doi.org/10.1016/j.cageo.2016.06.015

Chen N, Kemeny J, Jiang Q, Pan Z (2017) Automatic extraction of blocks from 3D point clouds of fractured rock. Comput Geosci 109:149–161. https://doi.org/10.1016/j.cageo.2017.08.013

Chen R, Lin C, Chen Y, He T, Fei L (2015) Detecting and characterizing active thrust fault and deep-seated landslides in dense forest areas of Southern Taiwan using airborne LiDAR DEM. REMOTE Sens 7:15443–15466. https://doi.org/10.3390/rs71115443

Chen W, Su L, Chen X, Huang Z (2023) Rock image classification using deep residual neural network with transfer learning. Front Earth Sci 10

Daghigh H, Tannant DD, Daghigh V, Lichti DD, Lindenbergh R (2022) A critical review of discontinuity plane extraction from 3D point cloud data of rock mass surfaces. Comput Geosci 169:105241. https://doi.org/10.1016/j.cageo.2022.105241

de Oliveira L, Lassiter H, Wilkinson B, Whitley T, Ifju P, Logan S, Peter G, Vogel J, Martin T (2021) Moving to automated tree inventory: comparison of UAS-derived lidar and photogrammetric data with manual ground estimates. REMOTE Sens. 13. https://doi.org/10.3390/rs13010072

Deibe D, Amor M, Doallo R (2020) Big data geospatial processing for massive aerial LiDAR datasets. REMOTE Sens 12. https://doi.org/10.3390/rs12040719

Esmaeilzadeh A, Shahriar K (2019) Optimized fuzzy cmeans—fuzzy covariance—fuzzy maximum likelihood estimation clustering method based on deferential evolutionary optimization algorithm for identification of rock mass discontinuities sets. Period Polytech Civ Eng 63:674–686. https://doi.org/10.3311/PPci.13885

Fekete G, Treinish LA (1990) Sphere quadtrees: a new data structure to support the visualization of spherically distributed data. In: Extracting meaning from complex data: processing, display, interaction. Presented at the extracting meaning from complex data: processing, display, interaction, SPIE, pp 242–253. https://doi.org/10.1117/12.19991

Ferrero A, Umili G, Vagnon F (2016) Analysis of discontinuity data obtained with remote sensing tools to generate input for EC7 design. In: Ulusay R, Aydan O, Gercek H, Hindistan M, Tuncay E (eds). Presented at the rock mechanics and rock engineering: from the past to the future, vol 2, pp 1115–1120

Ge Y, Tang H, Xia D, Wang L, Zhao B, Teaway JW, Chen H, Zhou T (2018) Automated measurements of discontinuity geometric properties from a 3D-point cloud based on a modified region growing algorithm. Eng Geol 242:44–54. https://doi.org/10.1016/j.enggeo.2018.05.007

Gigli G, Lombardi L, Carla T, Beni T, Casagli N (2022) A method for full three-dimensional kinematic analysis of steep rock walls based on high-resolution point cloud data. Int J Rock Mech Min Sci 157. https://doi.org/10.1016/j.ijrmms.2022.105178

Giordan D, Hayakawa Y, Nex F, Remondino F, Tarolli P (2018) Review article: the use of remotely piloted aircraft systems (RPASs) for natural hazards monitoring and management. Nat Hazards Earth Syst Sci 18:1079–1096. https://doi.org/10.5194/nhess-18-1079-2018

Griwodz C, Gasparini S, Calvet L, Gurdjos P, Castan F, Maujean B, Lanthony Y, de Lillo G (2021) alicevision meshroom: an open-source 3D reconstruction pipeline, In: 12th ACM multimedia systems conference (MMSys 2021). ACM: Association for Computing Machinery, Istanbul, Turkey, pp 241–247. https://doi.org/10.1145/3458305.3478443

Gu W, Bai S, Kong L (2022) A review on 2D instance segmentation based on deep neural networks. Image vis Comput 120:104401. https://doi.org/10.1016/j.imavis.2022.104401

Guo J, Zheng J, Lü Q, Xiao Z, Liu T (2022) An analysis of trace information of different-shaped fracture networks having a same fracture intensity (P32). KSCE J Civ Eng 26:4265–4275. https://doi.org/10.1007/s12205-022-1165-3

Hafiz AM, Bhat GM (2020) A survey on instance segmentation: state of the art. Int J Multimed Inf Retr 9:171–189. https://doi.org/10.1007/s13735-020-00195-x

Hammah R, Curran J (2000) Validity measures for the fuzzy cluster analysis of orientations. IEEE Trans Pattern Anal Mach Intell 22:1467–1472. https://doi.org/10.1109/34.895981

He K, Gkioxari G, Dollár P, Girshick R (2018) Mask R-CNN

Herrera G, Tomas R, Vicente F, Lopez-Sanchez J, Mallorqui J, Mulas J (2010) Mapping ground movements in open pit mining areas using differential SAR interferometry. Int J Rock Mech Min Sci 47:1114–1125. https://doi.org/10.1016/j.ijrmms.2010.07.006

Jaboyedoff M, Metzger R, Oppikofer T, Couture R, Derron M, Locat J, Turmel D (2007) New insight techniques to analyze rock-slope relief using DEM and 3D-imaging cloud points: COLTOP-3D software. In: Eberhardt E, Stead D, Morrison T (eds) Rock mechanics: meeting society's challenges and demands. Taylor & Francis, pp 61–68. https://doi.org/10.1201/NOE0415444019-c8

Jaboyedoff M, Oppikofer T, Abellan A, Derron M, Loye A, Metzger R, Pedrazzini A (2012) Use of LIDAR in landslide investigations: a review. Nat Hazards 61:5–28. https://doi.org/10.1007/s11069-010-9634-2

Jimenez-Rodriguez R, Sitar N (2006) A spectral method for clustering of rock discontinuity sets. Int J Rock Mech Min Sci 43:1052–1061. https://doi.org/10.1016/j.ijrmms.2006.02.003

Jones L, Hobbs P (2021) The application of terrestrial LiDAR for geohazard mapping, monitoring and modelling in the british geological survey. Remote Sens. 13. https://doi.org/10.3390/rs13030395

Khaloo A, Lattanzi D (2017) Robust normal estimation and region growing segmentation of infrastructure 3D point cloud models. Adv Eng Inform 34:1–16. https://doi.org/10.1016/j.aei.2017.07.002

Kong D (2021) Development and application of UAV-SfM photogrammetry for quantitative characterization of rock mass discontinuities. Int J Rock Mech Min Sci 19

Kong D, Wu F, Saroglou C (2020) Automatic identification and characterization of discontinuities in rock masses from 3D point

clouds. Eng Geol 265:105442. https://doi.org/10.1016/j.enggeo.2019.105442

Lato M, Kemeny J, Harrap R, Bevan G (2013) Rock bench: Establishing a common repository and standards for assessing rockmass characteristics using LiDAR and photogrammetry. Comput Geosci 50:106–114. https://doi.org/10.1016/j.cageo.2012.06.014

Laux D, Henk A (2015) Terrestrial laser scanning and fracture network characterisation—perspectives for a (semi-) automatic analysis of point cloud data from outcrops. Z Dtsch Ges Geowiss 166:99–118. https://doi.org/10.1127/1860-1804/2015/0089

Lee Y-S, Koo H-S, Jeong C-S (2006) A straight line detection using principal component analysis. Pattern Recognit Lett 27:1744–1754. https://doi.org/10.1016/j.patrec.2006.04.016

Li X, Chen J, Zhu H (2016) A new method for automated discontinuity trace mapping on rock mass 3D surface model. Comput Geosci 89:118–131. https://doi.org/10.1016/j.cageo.2015.12.010

Li X, Chen Z, Chen J, Zhu H (2019) Automatic characterization of rock mass discontinuities using 3D point clouds. Eng Geol 259. https://doi.org/10.1016/j.enggeo.2019.05.008

Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Zitnick CL, Dollár P (2015) Microsoft COCO: common objects in context

Liu L, Xiao J, Wang Y (2019) Major orientation estimation-based rock surface extraction for 3D rock-mass point clouds. Remote Sens. 11. https://doi.org/10.3390/rs11060635

Liu Y, Chen J, Tan C, Zhan J, Song S, Xu W, Yan J, Zhang Y, Zhao M, Wang Q (2022) Intelligent scanning for optimal rock discontinuity sets considering multiple parameters based on manifold learning combined with UAV photogrammetry. Eng Geol 309. https://doi.org/10.1016/j.enggeo.2022.106851

Papandreou G, Zhu T, Kanazawa N, Toshev A, Tompson J, Bregler C, Murphy K (2017) Towards accurate multi-person pose estimation in the wild. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Honolulu, HI, pp 3711–3719. https://doi.org/10.1109/CVPR.2017.395

Priest SD (1993) Discontinuity analysis for rock engineering. Springer

Priest SD (1985) Hemispherical projection methods in rock mechanics. HarperCollins Publishers Ltd, Boston

Qiao L, Deng N-F, Li Q-W, Zhang Q-L, Hao J-W (2022) Multi-Crack-Net: a fast segmentation and quantification combined method for rock discontinuity traces by capturing an image. Geofluids 2022:1832662. https://doi.org/10.1155/2022/1832662

Raguram R, Chum O, Pollefeys M, Matas J, Frahm J (2013) USAC: a universal framework for random sample consensus. IEEE Trans PATTERN Anal Mach Intell 35:2022–2038. https://doi.org/10.1109/TPAMI.2012.257

Ran X, Xue L, Zhang Y, Liu Z, Sang X, He J (2019) Rock classification from field image patches analyzed using a deep convolutional neural network. Mathematics 7:755. https://doi.org/10.3390/math7080755

Rathje E, Franke K (2016) Remote sensing for geotechnical earthquake reconnaissance. SOIL Dyn Earthq Eng 91:304–316. https://doi.org/10.1016/j.soildyn.2016.09.016

Riquelme A, Tomas R, Cano M, Pastor J, Abellan A (2018) Automatic mapping of discontinuity persistence on rock masses using 3D point clouds. Rock Mech Rock Eng 51:3005–3028. https://doi.org/10.1007/s00603-018-1519-9

Riquelme AJ, Abellán A, Tomás R, Jaboyedoff M (2014) A new approach for semi-automatic rock mass joints recognition from 3D point clouds. Comput Geosci 68:38–52. https://doi.org/10.1016/j.cageo.2014.03.014

Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344:1492–1496. https://doi.org/10.1126/science.1242072

Rousseeuw PJ (1987) Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Ruiz Pereira S, Marquardt C, Beriain E, Lambert F (2021) Permafrost evolution in a mountain catchment near Santiago de Chile. J South Am Earth Sci 109:103293. https://doi.org/10.1016/j.jsames.2021.103293

Singh SK, Banerjee BP, Lato MJ, Sammut C, Raval S (2022a) Automated rock mass discontinuity set characterisation using amplitude and phase decomposition of point cloud data. Int J Rock Mech Min Sci 152. https://doi.org/10.1016/j.ijrmms.2022.105072

Singh SK, Banerjee BP, Lato MJ, Sammut C, Raval S (2022b) Automated rock mass discontinuity set characterisation using amplitude and phase decomposition of point cloud data. Int J Rock Mech Min Sci 152:105072. https://doi.org/10.1016/j.ijrmms.2022.105072

Singh SK, Raval S, Banerjee BP (2021) Automated structural discontinuity mapping in a rock face occluded by vegetation using mobile laser scanning. Eng Geol 285:106040. https://doi.org/10.1016/j.enggeo.2021.106040

Smith AR (1978) Color gamut transform pairs. ACM SIGGRAPH Comput Graph 12:12–19. https://doi.org/10.1145/965139.807361

Smith JV, Holden L (2021) Rock slope kinematic instability controlled by large-scale variation of basalt column orientation. Bull Eng Geol Environ 80:239–250. https://doi.org/10.1007/s10064-020-01917-5

Sturzenegger M, Stead D (2009) Quantifying discontinuity orientation and persistence on high mountain rock slopes and large landslides using terrestrial remote sensing techniques. Nat Hazards Earth Syst Sci 9:267–287. https://doi.org/10.5194/nhess-9-267-2009

Vo A, Linh T, Laefer D, Bertolotto M (2015) Octree-based region growing for point cloud segmentation. ISPRS J Photogramm Remote Sens 104:88–100. https://doi.org/10.1016/j.isprsjprs.2015.01.011

Wada K (2023) Labelme: image polygonal annotation with python. https://doi.org/10.5281/zenodo.5711226

Wang X, Zou L, Shen X, Ren Y, Qin Y (2017) A region-growing approach for automatic outcrop fracture extraction from a three-dimensional point cloud. Comput Geosci 99:100–106. https://doi.org/10.1016/j.cageo.2016.11.002

Wang Y, Feng H, Delorme F, Engin S (2013) An adaptive normal estimation method for scanned point clouds with sharp features. Comput-AIDED Des 45:1333–1348. https://doi.org/10.1016/j.cad.2013.06.003

Wu W, Zhang K, Zhu H (2020) A fast automatic extraction method for rock mass discontinuity orientation using fast k-means++ and fast silhouette based on 3D point cloud. IOP Conf Ser Earth Environ Sci 570:052075. https://doi.org/10.1088/1755-1315/570/5/052075

Xie XL, Beni G (1991) A validity measure for fuzzy clustering. IEEE Trans Pattern Anal Mach Intell 13:841–847. https://doi.org/10.1109/34.85677

Yan J, Chen J, Zhan J, Song S, Zhang Y, Zhao M, Liu Y, Xu W (2022) Automatic identification of rock discontinuity sets using modified agglomerative nesting algorithm. Bull Eng Geol Environ. https://doi.org/10.1007/s10064-022-02724-w

Yan W, Shaker A, El-Ashmawy N (2015) Urban land cover classification using airborne LiDAR data: a review. Remote Sens Environ 158:295–310. https://doi.org/10.1016/j.rse.2014.11.001

Zaidi SSA, Ansari MS, Aslam A, Kanwal N, Asghar M, Lee B (2022) A survey of modern deep learning based object detection models. Digit Signal Process 126:103514. https://doi.org/10.1016/j.dsp.2022.103514

Zhang K, Wu W, Zhu H, Zhang L, Li X, Zhang H (2020) A modified method of discontinuity trace mapping using three-dimensional point clouds of rock mass surfaces. J Rock Mech Geotech Eng 12:571–586. https://doi.org/10.1016/j.jrmge.2019.10.006

Zhang L, Einstein H (1998) Estimating the mean trace length of rock discontinuities. Rock Mech Rock Eng 31:217–235. https://doi.org/10.1007/s006030050022

Zhang L, Einstein HH (2000) Estimating the intensity of rock discontinuities. Int J Rock Mech Min Sci 37:819–837. https://doi.org/10.1016/S1365-1609(00)00022-8

Zhang P, Du K, Tannant DD, Zhu H, Zheng W (2018) Automated method for extracting and analysing the rock discontinuities from point clouds based on digital surface model of rock mass. Eng Geol 239:109–118. https://doi.org/10.1016/j.enggeo.2018.03.020

Zhang Q, Wang X, He L, Tian L (2021) Estimation of fracture orientation distributions from a sampling window based on geometric probabilistic method. Rock Mech Rock Eng 54:3051–3075. https://doi.org/10.1007/s00603-021-02431-0

Zhao R, Shi S, Li S, Guo W, Zhang T, Li X, Lu J (2023) Deep learning for intelligent prediction of rock strength by adopting measurement while drilling data. Int J Geomech 23:04023028. https://doi.org/10.1061/IJGNAI.GMENG-8080

Zhu H, Wu W, Chen J, Ma G, Liu X, Zhuang X (2016) Integration of three dimensional discontinuous deformation analysis (DDA) with binocular photogrammetry for stability analysis of tunnels in blocky rockmass. Tunn Undergr Space Technol 51:30–40. https://doi.org/10.1016/j.tust.2015.10.012

Zou Z, Chen K, Shi Z, Guo Y, Ye J (2023) Object detection in 20 years: a survey