The Visual Computer (2020) 36:669–681
https://doi.org/10.1007/s00371-019-01648-z

ORIGINAL ARTICLE

An automatic 3D registration method for rock mass point clouds based on plane detection and polygon matching

Liang Hu[1] · Jun Xiao[1] · Ying Wang[1]

Published online: 2 April 2019

© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

Point cloud registration is an essential step in the process of 3D reconstruction. Considering that the surface of rock mass is complex and mainly composed of planes, in this paper, we introduce a novel and automatic 3D registration method for rock mass point clouds based on plane detection and polygon matching. Firstly, planes are detected from rock mass point clouds by an efficient tripe-region growing method, and then, the corresponding polygons are calculated by concave hull method. Secondly, PCA-based polygon matching procedure is used for coarse registration. Finally, ICP method is applied to fine registration. The performance of this method was tested in different rock mass point clouds. Compared with the existing methods, the proposed method demonstrates a reliable and stable solution for accurately registering in rock mass scenes.

Keywords Automatic registration · Rock mass · Plane detection · Polygon matching

1 Introduction

Terrestrial laser scanning (TLS) can deliver dense and accurate 3D point clouds [1,2] and is widely used in many scientific fields including 3D model reconstruction [3], 3D object recognition [4,5] and cultural heritage management [6]. TLS has limited measurement range, and thus, in order to obtain full coverage of a scene multiple scans from different viewpoints are necessary. Then, multiple point clouds must be transformed into a common coordinate system in a registration step. Hence, point cloud registration [7] is a key component in processing 3D point clouds data.

In rock engineering, the technique of laser scanning can rapidly obtain the spatial information about rock mass as point clouds. The 3D model of rock mass reconstructed from point clouds is the initial input for numerical analysis methods, which can be efficiently used for the analysis of rock stability [8,9]. As the important step of 3D reconstruction, point cloud registration is also attracted increasing attention in the field of rock engineering.

The purpose of registration is to find a transform matrix between two point clouds scanned from different viewpoints. Although several methods have been proposed for automatic registration [10], they still have more or less difficulties in dealing with rock mass point clouds due to the unique geometrical characteristics of rock mass, for example, different orientations of discontinuities, irregular shapes and different roughness. As shown in Fig. 1, rock mass discontinuities [11] which are represented by planes are important features in rock engineering. Recognition discontinuities can not only be used for registration but also provide initial primitives (planes) for 3D reconstruction of rock mass. Motivated by this significant feature, plane information is more suitable for registration between rock mass point clouds than other features.

In this paper, we focus on 3D rigid registration for rock mass point clouds. Firstly, planes are extracted in rock mass point clouds using an efficient plane detection method. Then, polygon matching procedure is used for coarse registration. Finally, fine registration is carried out using ICP method. All processes are optimized for the rock mass scenes. Our work makes the following contributions:

– An efficient triple-region growing method has been proposed for plane detection in rock mass point clouds. The advantages of different kinds of region growing (RG) are exploited in this algorithm. This integrated strategy exhibits better adaptability to complex rock structures and moreover achieves higher accuracy than existing methods.

✉ Jun Xiao
xiaojun@ucas.ac.cn

[1] University of Chinese Academy of Science, Beijing, China

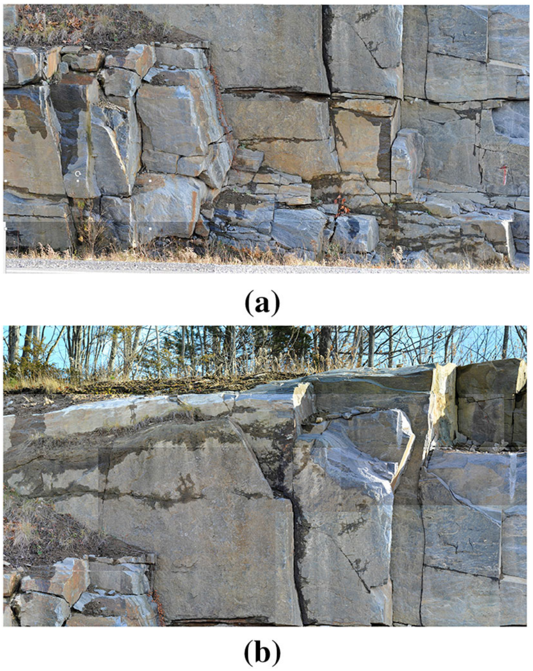


Fig. 1 Examples of natural rock mass. The outer surfaces of naturally formed rock masses are often discontinuities, comprised mainly of planes

- Considering that the surface of the rock mass is mostly composed of planes and these planes usually have different shapes, a fast coarse registration method has been proposed for rock mass scenes based on plane detection and polygon matching. In the final step, iterative closest point (ICP) method is performed for fine registration. This strategy is fast and more suitable for registration between rock mass point clouds.
- We conduct experiments on different rock mass point clouds to evaluate the proposed method. The experimental result shows that our method can achieve better performance than the other compared methods on registration between rock mass point clouds.

The paper is organized as follows. The related work is introduced first in Sect. 2. Then, the details of proposed method are thoroughly discussed in Sect. 3. Section 4 includes a discussion and the experimental results. Finally, we summarize this paper and introduce the future work in Sect. 5.

2 Related work

2.1 Registration

3D registration is a crucial step in 3D model reconstruction [12]. Point cloud registration can be divided into coarse registration and fine registration. Coarse registration is used to determine approximation of transformation between two point clouds. Fine registration focuses on obtaining more accurate transformation after coarse registration procedure.

There has been a large body of work surrounding automatic coarse registration; most of them are feature-based methods. Coarse registration is an approach to calculate a rough initial transformation to increase the performance of fine registration. Rusu et al. [13] proposed point feature histograms (FPFH) for registration. In this method, a histograms-based descriptor is computed for each point within point cloud, and then, the correspondences between points from different shapes are found based on this feature. Yang et al. [14] used the standard scale-invariant feature transform (SIFT) detector to calculate the transformation in point clouds. Guo et al. [15] proposed a rotational projection statistics (RoPS) feature for point cloud registration and 3D object recognition. In addition, 4-points congruent sets [16,17] and spatial curves [18] are also commonly used in coarse registration.

Xian et al. [19] produced an imaged-based method for registration. They first convert point clouds to images based on spherical projection. The method of SIFT was used to search corresponding relationship in images. Then, the correspondences were applied to coarse registration in 3D point clouds. Recently, deep-learning based method has been proposed for 3D registration. Elbaz et al. [20] use a deep neural network auto-encoder for registration in point clouds.

The iterative closest point (ICP) [21] is one of the most extensive access to fine registration. As ICP is a local optimization, this method requires a suitable initial alignment to reduce the possibility of local minimum. Coarse registration is designed to overcome this limitation by determine approximation of transformation between two point clouds. What is more, some variants of ICP are proposed to improve the performance. Yang et al. [22] proposed Go-ICP to extend ICP with global optimal guarantee by adopting a branch and bound strategy. Pomerleau et al. [23] developed a fast ICP algorithm for real-time registration to adapt to SLAM scenarios.

2.2 Plane detection

Researchers have proposed several algorithms on 3D plane detection. According to the working principle, Hough transform (HT), random sample consensus (RANSAC) and region growing (RG) are the most popular approaches to detect planes in point clouds. This section discusses these algorithms and their various optimizations which are intended to accelerate plane detection in point clouds.

HT is a robust method, and it uses the correspondence between the object space and the parameter space to detect parameterized models. The high computation cost is the main

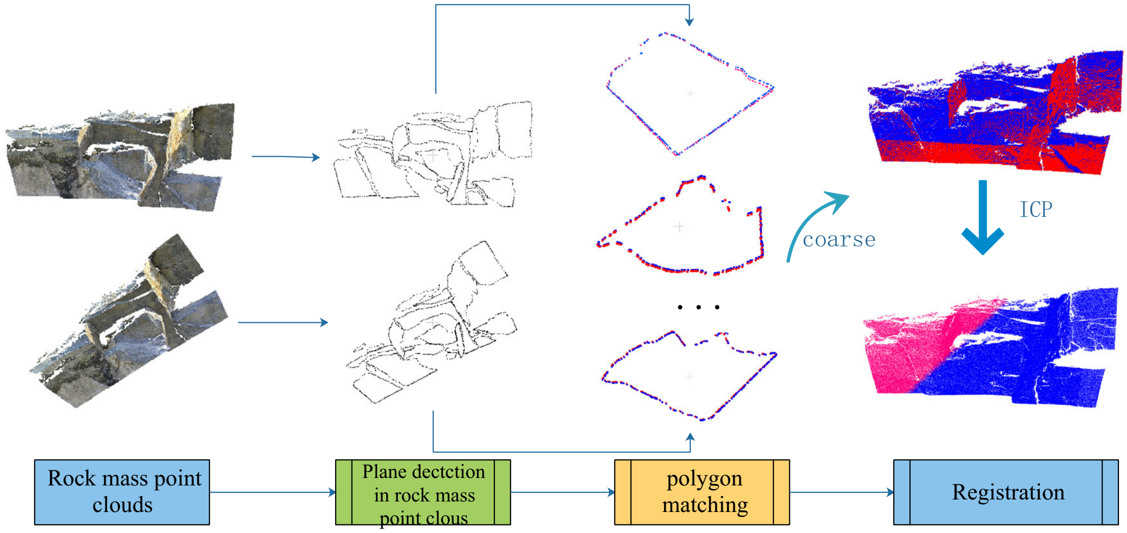


Fig. 2 The framework of the proposed method. Firstly, planes and polygons are extracted from rock mass point clouds. Secondly, polygon pairs can be determined on 2D coordinate system. Then, the centroid of plane point sets corresponding to polygon pairs is used as point pairs for coarse registration. Finally, ICP method is used for fine registration

problem of this method. Borrmann et al. [24] compared four different optimized HT using a new designed sphere accumulator. The experimental results present that randomized Hough transform (RHT) outperforms all the other HT methods for detecting an unknown number of planes in point clouds. Limberger and Oliveira [25] proposed a 3D Kernel-based Hough transform (KHT) based on Octree and principal component analysis (PCA) for real-time plane detection.

RANSAC is another model-based algorithm for detection of geometric primitives, and it is an interactive method to detect an accurate plane based on probability. Schnabel et al. [26] proposed two optimizations to improve the performance of RANSAC: localized sampling strategy based on Octree and optimized score function based on statistics. This method is widely used for plane detection in the field of building modeling.

RG [27] is a segmentation method based on neighborhood information. A seed region is first picked randomly in this method, and then, a local search is performed to expand regions when the neighbors satisfy some conditions. Therefore, different seed regions and growing rules influence the performance of RG greatly. In general, the distance from point to plane and the mean square error (MSE) were frequently adopted as growing rules. Traditional RG usually used a single point as growing unit. Poppoinga et al. [28] proposed an optimized RG method using an incremental version for plane parameters computation. Holz and Behnke [29] extended the point normals captured from point cloud as growing rules. Apart from this rules, Xiao et al. [30] designed an optimized method for coplanarity test. Wang et al. [31] improved the performance by an optimized MSE calculation in the field of autonomous navigation.

It is noted that the growing unit is a single point in the above traditional RG methods, which is inefficiency in processing massive point clouds. Xiao et al. [5] presented a voxel-based RG for plane detection in range images. In this approach, point cloud is divided into small voxel and each voxel satisfied with some conditions is regarded as growing unit. Gomes et al. [32] presented a PCA-based algorithm to detect planar and near-planar surfaces automatically. Oesau et al. [33] used K-nearest neighbor (KNN) to select seed points.

3 Methodology

This section presents the details of our methodology that is proposed for registration on rock mass point clouds. The framework of the proposed method is illustrated in Fig. 2. The method consists of three sequent steps: plane detection, polygon matching and registration. The functionality of each step will be briefly explained next.

3.1 Plane detection

Several traditional methods have been proposed to detect plane in point clouds; however, few methods can be directly used for rocky points due to the unique geometrical characteristics of rock mass, such as irregular shapes, wide variations in elevations, large variations in spatial extents and the different roughness. A novel triple-RG method is proposed in order to detect planes in rock mass point clouds efficiently. The pipeline of this method is shown in Fig. 3. Because the normal vector of each point plays an important role in detect-

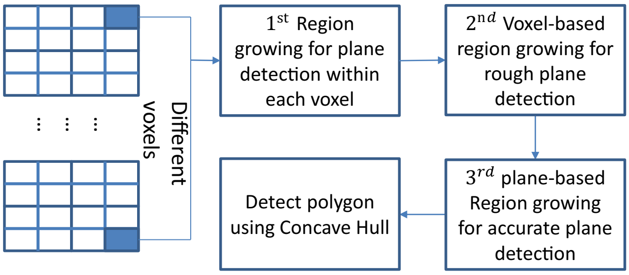

Different voxels
1st Region growing for plane detection within each voxel
2nd Voxel-based region growing for rough plane detection
3rd plane-based Region growing for accurate plane detection
Detect polygon using Concave Hull

Fig. 3 The pipeline of plane detection in rock mass point cloud. Firstly, point cloud is divided into small voxels. Secondly, a point-based RG (first) is used in each voxel for coplanarity test to generate growing units. Then, a voxel-based RG (second) is used to detect planes based on the growing units. Finally, a plane-based RG (3rd) is used to process the remaining points for detecting accurate planes

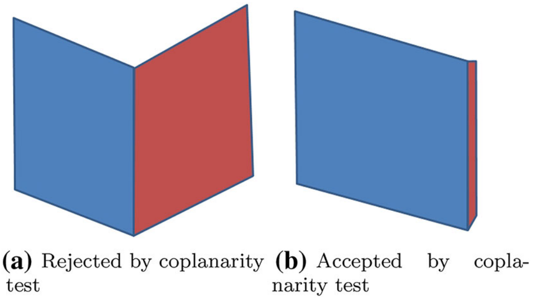

(a) Rejected by coplanarity test **(b)** Accepted by coplanarity test

Fig. 4 Example of traditional methods for coplanarity test. **a** and **b** Point clouds within a voxel, which contains two different planes (red and blue parts). The rejection of **a** will make the final plane incomplete; the acceptance of **b** will make the final plane inaccurate

ing planes with different orientation, the normal information should be pre-calculated.

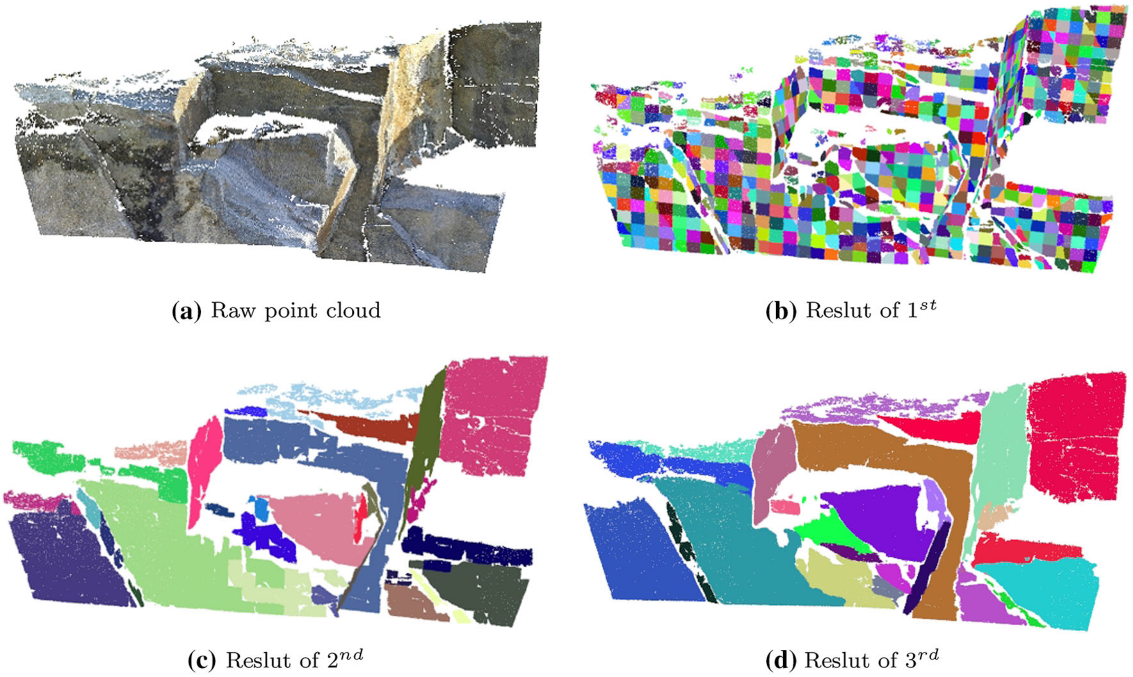
3.1.1 Triple-region growing

The engineering point clouds obtained from laser scanning are always massive. Thus, voxel-based RG (second) is chosen in our method, which is more efficient for plane detection than other methods. In this method, point cloud is first divided into small voxels according to the bounding box and specified resolution. Then, these voxels are regarded as growing units and clustered into completed planes by RG. However, the performance of this method is greatly influenced by the quality of seed regions. In order to detect planes accurately, we only choose the voxels which pass the coplanarity test as growing units. Several methods have been applied to coplanarity test. Xiao et al. [5] used the minimum eigenvalue calculated by the covariance matrix. If the points within a voxel are coplanar, the minimum eigenvalue is getting closer to zero. Gigli et al. [34] estimated the least-squares plane aiming at each voxel and used the distance variance between the point and fitting plane for coplanarity test. These methods above are efficiency but not accuracy. As shown in Fig. 4, the red and blue parts represent two different planes within a voxel. If a voxel contains two or more similar plane (Fig. 4a), this voxel should be removed according to the traditional coplanarity test, which will make the final plane incomplete. If a voxel contains one big planes and some small planes (Fig. 4b), this voxel may be accepted, which will make the final plane inaccurate. This two situation often occurs at the boundary of planes and will reduce the accuracy of voxel-based RG.

For solving this problem, we add two RG procedures. A point-based RG (1st) is used for coplanarity test, which can detect a best plane as growing unit within each voxel. RG acquires high-quality and accurate growing unit efficiently, which can improve the performance of voxel-based RG. After the procedure of voxel-based RG, the completed planes can be roughly detected, but some information loss exists at the boundary of planes. Because one voxel may not contain one plane, the smaller plane points should be removed after coplanarity test. Meanwhile, because of the different degrees of roughness in rock mass point clouds, some points with large roughness may be discarded as well. Thus, we use a plane-based RG (third) to deal with the remaining points. As the information loss often appears at interiors and the boundary of planes, we only consider the points located on these regions. In triple RG, we use the distance and the angle between two planes (or between points and planes) as growing rules. The angle between two planes constrains the orientation of the plane. The distance between two planes constrains the spacing between planes. Larger thresholds can adapt to complex environments, but will cause inaccuracy. The details of triple-region growing are shown in Fig. 5. After the procedure of triple-region growing, planes can be detected from rock mass point clouds efficiently and accurately.

Algorithm 1 shows the common RG method to detect planes. All the three RG methods above can use Algorithm 1 to detect planes by setting different growing units, growing rules and seed selection strategy, respectively.
The process of triple-RG method is as follows:

(1) Point cloud is first divided into small voxels according to the bounding box and specified resolution, and the neighborhood information is built at the same time. The resolution should be set manually according to the point density and scale of point cloud. Low-resolution settings may cause unfavorable results that there are not enough points to fit the plane in each voxel. Large resolution settings may generate negative influences that there are too many planes within one voxel to reduce accuracy. According to our experience, the minimum resolution is more than ten times the average distance of points.
(2) A point-based RG (1st) is used in each voxel for coplanarity test to generate growing units. In each voxel,


(a) Raw point cloud

(b) Reslut of 1^{st}

(c) Reslut of 2^{nd}

(d) Reslut of 3^{rd}

Fig. 5 The procedure of triple-region growing. **a** Raw rock mass point cloud; **b** result of point-based RG (first) for generate growing units in each voxel; **c** result of voxel-based RG (second) for detecting initial planes; **d** result of plane-based RG (third) to process remaining points around initial planes for detecting accurate planes

Algorithm 1 Algorithm for RG

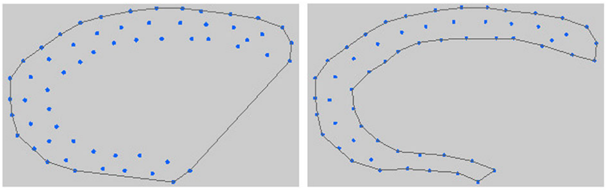
```
$seed$ = select seeds from $growing\_unit\_set$;
$Plane$ = calculate the plane parameter of $seed$;
push the seeds in $seed\_queue$;
while $seed\_queue$ is not empty do
    $sp$ = pop seeds from $seed\_queue$;
    while all the neighbors of $sp$ are processed do
        $np$ = select one neighbor of $sp$ do not be processed.
        $dis$ = calculate the distance between $Plane$ and $np$;
        $angle$ = calculate the angle between the normal vector of $Plane$ and $np$;
        if $dis < d_{min}$ and $angle < \theta_{min}$ then
            push the $np$ in $seed\_queue$;
            $Plane = Plane + np$;
            recalculate the parameter of $Plane$;
        end if
    end while
end while
return $Plane$;
```

we choose points close to the center of the voxel as seed points. Then, we choose 3 non-coplanar points to fit plane $\mathcal{P}$ and begin to grow. Suppose that $Ax + By + Cz + D = 0$ represents the plane $\mathcal{P}$ equation. $p_i = (p_{xi}, p_{yi}, p_{zi})$ represents the point within the voxel and $n_i = (n_{xi}, n_{yi}, n_{zi})$ is the unit point normal vector of p_i. The distance and the angle between neighborhood points and plane $\mathcal{P}$ are used as growing rules. If the distance between the point p_i and the $\mathcal{P}$ (Eq. 1) is less than the threshold $d_{\min}$, and the angle between the plane normal vector (A, B, C) and point vector n_i (Eq. 2) is less than the threshold $\theta_{\min}$, the point p_i can be added to the plane $\mathcal{P}$. After all the points within this voxel have been calculated, one completed plane can be detected. This process repeats 3 times, and we choose the plane containing the largest number of points as growing unit in each voxel. According to our experience, the distance threshold $d_{\min}$ does not exceed 0.1m and the angle threshold $\theta_{\min}$ does not exceed 15°. The result of this process is shown in Fig. 5b.

$$\frac{\left|Ap_{xi} + Bp_{yi} + Cp_{zi} + D\right|}{\sqrt{A^2 + B^2 + C^2}} \leq d_{\min} \tag{1}$$

$$\arccos\left(\frac{\left|An_{xi} + Bn_{yi} + Cn_{zi}\right|}{\sqrt{A^2 + B^2 + C^2}}\right) \leq \theta_{\min} \tag{2}$$

(3) A voxel-based RG (second) is used to detect planes based on the growing units. This process repeats iteratively relied on neighborhood information. Each iteration can detect one completed plane and contain three steps: firstly, plane which contains the most number of points is selected as seed plane. The distance and the angle between two plane are used as growing rules. Then, the neighbors of seed plane satisfied with the growing rules are added together with the seed regions.



(a) Result of Convex Hull **(b)** Result of Concave Hull

Fig. 6 The method to compute polygon from point clouds

Finally, these neighbors become new seed regions to expand regions until no voxels are added as seed regions. The result of this process is shown in Fig. 5c.

(4) A plane-based RG (third) is used to detect accurate planes. Firstly, we select a plane as seed plane at one time from the plane set which is detected in the procedure of voxel-based RG. Then, the points around seed plane were verified to determine whether the point belongs to this plane. In this process, we choose the same growing rules as first RG. This process mainly focuses on the remaining points, which mainly exist in rough regions. Therefore, the threshold used in third RG should be set manually according to the roughness. Experiments demonstrate the threshold should be 50–100% greater than the threshold used in first RG. The result of this process is shown in Fig. 5d.

Algorithm 2 PCA-based Polygon Matching

```
Input: $poly\_source\_set$, $poly\_target\_set$;
Output: $relationship\_set$;
for each polygon $source_i$ in $poly\_source\_set$ from $i$ to $M$ do
    calculate the centroid ($O_s$) of $source_i$ (Eq3)
    translate the coordinate center to the centroid (Eq5)
    transform $source_i$ into a 2D coordinate (Eq6 - 7)
    for each polygon $target_j$ in $poly\_target\_set$ $j$ to $N$ do
        $S_{s_i}$ = the area of $source_i$
        $S_{t_j}$ = the area of $traget_j$
        $S_{s_i \bigcap t_j}$ = the area of intersections between $source_i$ and $target_j$
        if $Sr \geq 90\%$ (Eq8) then
            save the corresponding relationship in $relationship\_set$
            remove $target_j$ from $poly\_target\_set$
        end if
    end for
end for
return $relationship\_set$;
```

3.1.2 Polygon detection

Because point clouds are unorganized and do not contain topology information, it is inconvenient to find the correspondences between two point clouds. Thus, we use polygon based on planes which are detected after the procedure of triple-region growing to find the correspondences between two point clouds. Concave hull and convex hull are the most commonly used to compute polygon from point clouds. As shown in Fig. 6, the convex hull does not represent the boundaries of a given set of points well. Therefore, in this paper we use concave hull to detect polygon from point clouds. Concave hull method has been provided in the open-source point cloud library (PCL). The parameter of alpha in this method is determined as 0.5. Each polygon contains some ordered 3D points which approximately represent the boundary shape of a planar point cloud.

3.2 PCA-based polygon matching

In order to find the correspondences between point clouds, PCA-based polygon matching method is proposed in this paper. Algorithm 2 shows the PCA-based polygon matching. Firstly, we choose a polygon $source_i$ from $poly\_source\_set$ and then calculate the centroid O_s by Eq. 3. Then, we translate the coordinate center to the centroid by Eq. 4. Suppose that $source_i$ contains n points, p_i is the coordinates of one points in $source_i$.

$$O_s = \frac{1}{n}\sum_{i=1}^{n} p_i \tag{3}$$

$$p_i = p_i - O_s \quad (i \in [1, n]) \tag{4}$$

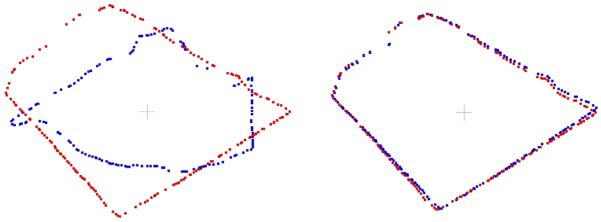
Secondly, we transform source into a 2D coordinate system using PCA. Because the planes and polygon detected from point clouds are of approximately two-dimensional shapes, finding correspondences between polygons in 2D coordinate system is more efficient and concise. As one of the most popular techniques to reduce dimension, PCA is applied. We assemble the covariance matrix Σ by Eq. 5. λ_j and v_j are the eigenvalues and eigenvectors of Σ ($\lambda_0 < \lambda_1 < \lambda_2$). Because the polygon is a 2D shape, the minimum eigenvalue λ_0 is close to zero. Suppose that the matrix $T = \{v_1, v_2\}$, then the 3D points in source can be transformed to 2D coordinate system by Eq. 7. In rock mass point clouds, most of the planes are irregular. Therefore, the two different orthogonal axes are easy to distinguish according to eigenvalues.

$$\Sigma = \frac{1}{n-1}\sum_{i=1}^{n} p_i p_i^T \tag{5}$$

$$\Sigma \cdot v_j = \lambda_j \cdot v_j, \, j \in \{0, 1, 2\} \tag{6}$$

$$p_{i\_2d} = p_i * T \quad (i \in [1, n]) \tag{7}$$

Finally, polygon matching can be implemented on 2D coordinate system. As the planes in rock mass point clouds are always irregular, the area of polygon is used to determine the correspondences between polygons. As the corresponding polygons may have similar number of points, we sequentially select one polygon $target_j$ from $poly\_target\_set$ if



(a) Incorrect matching **(b)** Correct matching

Fig. 7 Example of polygon matching

the difference between the number of points in $source_i$ and $target_j$ is less than 20%. Then, we calculate the area ($S_{s_i \bigcap t_j}$) of intersections between $source_i$ and $target_j$. Two polygons $source_i$ and $target_j$ are matched, if the similarity rate (Sr) satisfies Eq. 8. As shown in Fig. 7, red and blue parts represent two different polygons, and the area of polygon can help find the correct match. After traversing the polygons in $poly\_source\_set$, all correct correspondences will be determined.

$$Sr = \frac{2 * S_{s_i \bigcap t_j}}{S_{s_i} + S_{t_j}} \geq 90\% \tag{8}$$

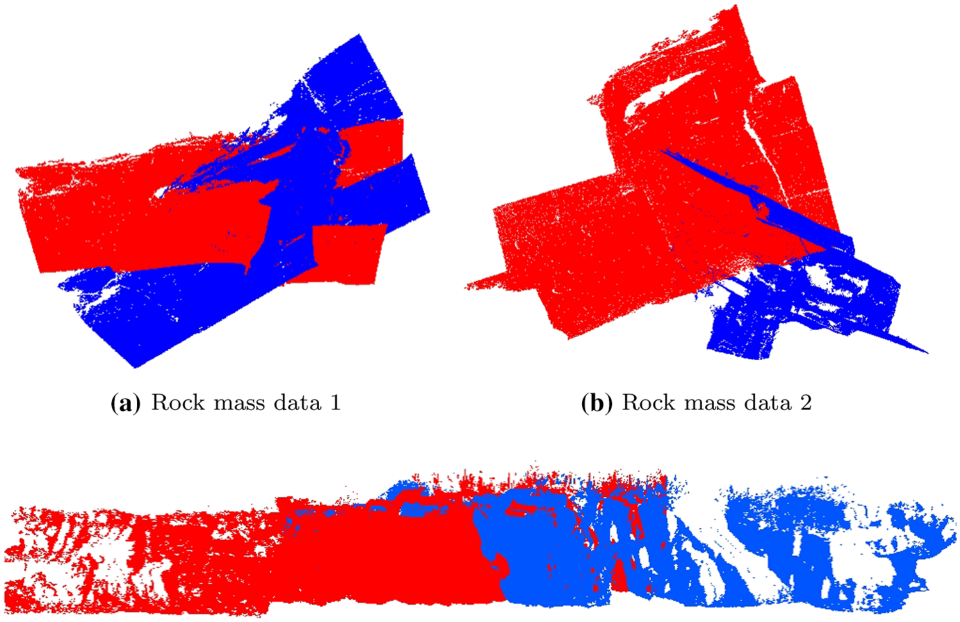
3.3 Registration

After finished determining the correspondences between polygons, a coarse registration will be implemented firstly in the registration procedure. In this paper, we focus on rigid registration, which has the 6DoF (6 degrees of freedom) transformation between the point clouds. Suppose that source point cloud $\mathcal{P}$ contains K key-points (p_k), and the corresponding points are q_k in target points $\mathcal{Q}$. The aim is to estimate the transformation $\mathcal{T} = [R|t]$, and then, we should minimize an alignment error measure ε by Eq. 9, where R (3DoF) is the rotation matrix and t is the translation matrix. Quaternion $\mathbf{q} = \{q_0, \mathbf{v}\} = \{q_0, q_1, q_2, q_3\}$ is often used to parameterize rotation matrix R by Eq. 10.

$$\varepsilon = \sum_{k=1}^{K} \|q_k - (Rp_k + t)\|^2 \tag{9}$$

$$R = \begin{bmatrix} 1-2q_2^2-2q_3^2 & 2q_1q_2-2q_0q_3 & 2q_1q_3+2q_0q_2 \\ 2q_1q_2+2q_0q_3 & 1-2q_1^2-2q_3^2 & 2q_2q_3+2q_0q_1 \\ 2q_1q_3-2q_0q_2 & 2q_2q_3+2q_0q_1 & 1-2q_1^2-2q_3^2 \end{bmatrix} \tag{10}$$

In order to estimate the transformation, we should find at least 3 corresponding points. Quaternion $\mathbf{q} = \{q_1, q_2, q_3, q_4\}$ is often used to parameterize rotation matrix R. In the proposed method, we use the best 3 point pairs to estimate accurate transformation and we can also use all the point pairs for more stable estimation. Firstly, three polygon pairs which have top three similarity rates are selected from $relationship\_set$. Then, we can find the corresponding plane point sets easily. Considering noise and equipment errors, we use the centroid of point sets as corresponding points, and then, the coarse transformation matrix can be estimated according to these points. Finally, a typical ICP method is performed for fine registration. ICP method has been provided in the open-source PCL.



(a) Rock mass data 1 **(b)** Rock mass data 2

(c) Rock mass data 3

Fig. 8 Rock mass point clouds data used for experimentation. Data 1 contain small-scale rock mass point clouds with small rotation and translation matrix. Data 2 contain middle-scale point clouds with large rotation matrix. Data 3 contain big-scale rough rock mass point clouds with partial overlap

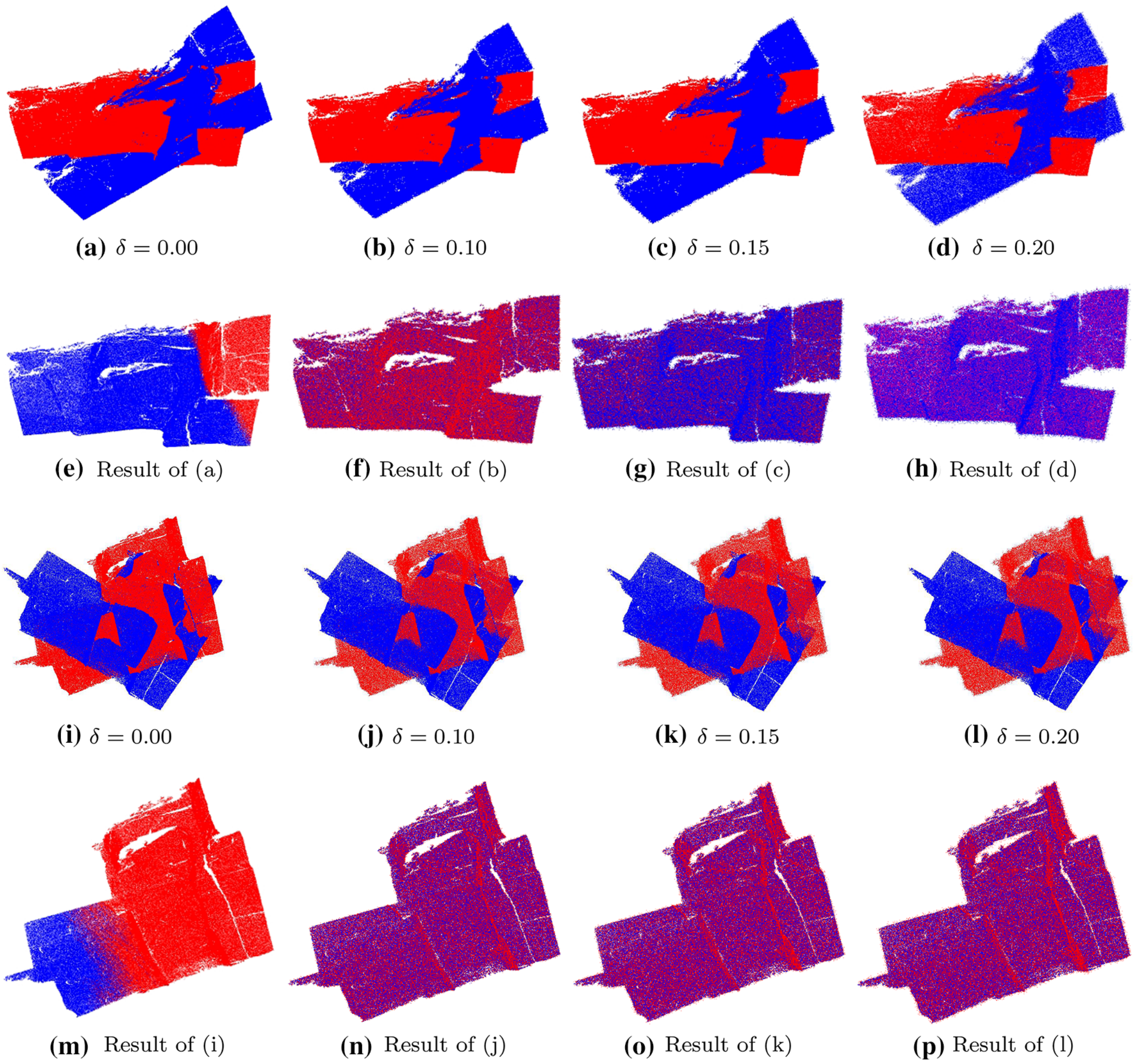


Fig. 9 Results achieved by using the proposed method with different Gaussian noises. **a**–**d** Data 1 with different Gaussian noises varying from $\delta = 0.00$ to $\delta = 0.20$; **e**–**h** registration results of **a**–**d** using the proposed method. **i**–**l** data 2 with different Gaussian noises varying from $\delta = 0.00$ to $\delta = 0.20$; **m**–**p** registration results of **i**–**l** using the proposed method

4 Experimental results and analysis

In order to test the performance of our method, three different rock mass data sets are assessed to validate the method, as shown in Fig. 8. All the rock mass data come from a public LiDAR data set at Rockbench [35]. Each of the rock mass data has two rock mass point clouds: source_cloud (red part) and target_cloud (blue part). Data 1 contain small-scale rock mass point clouds with small rotation and translation matrix. Data 2 contain middle-scale point clouds with large rotation matrix. Data 3 contain big-scale rough rock mass point clouds with partial overlap. Root mean square (RMS) error is computed for evaluating the performance. Firstly, the robust test is performed. Then, we compare the proposed method with other registration method to validate the performance of our method. For comparison, the terminating condition of ICP in all comparison methods is set as the difference value of REM error between two iterations which is less than 1.0e−5. Our technique has been implemented in C++, and all the experiments are carried out on PC with Intel I7-7700, 8 GB RAM.

4.1 Robust test

In this part, we use data 1 and data 2 to validate the robust of our method. We add Gaussian noise with three different standard deviations ($\delta \in \{0.10, 0.15, 0.20\}$) to target_cloud. Meanwhile, we rotate and translate the target_cloud to different initial positions for testing the robust of our method. The results are shown in Figs. 9 and 10. Because plane feature is a relatively stable feature in rock mass point clouds,

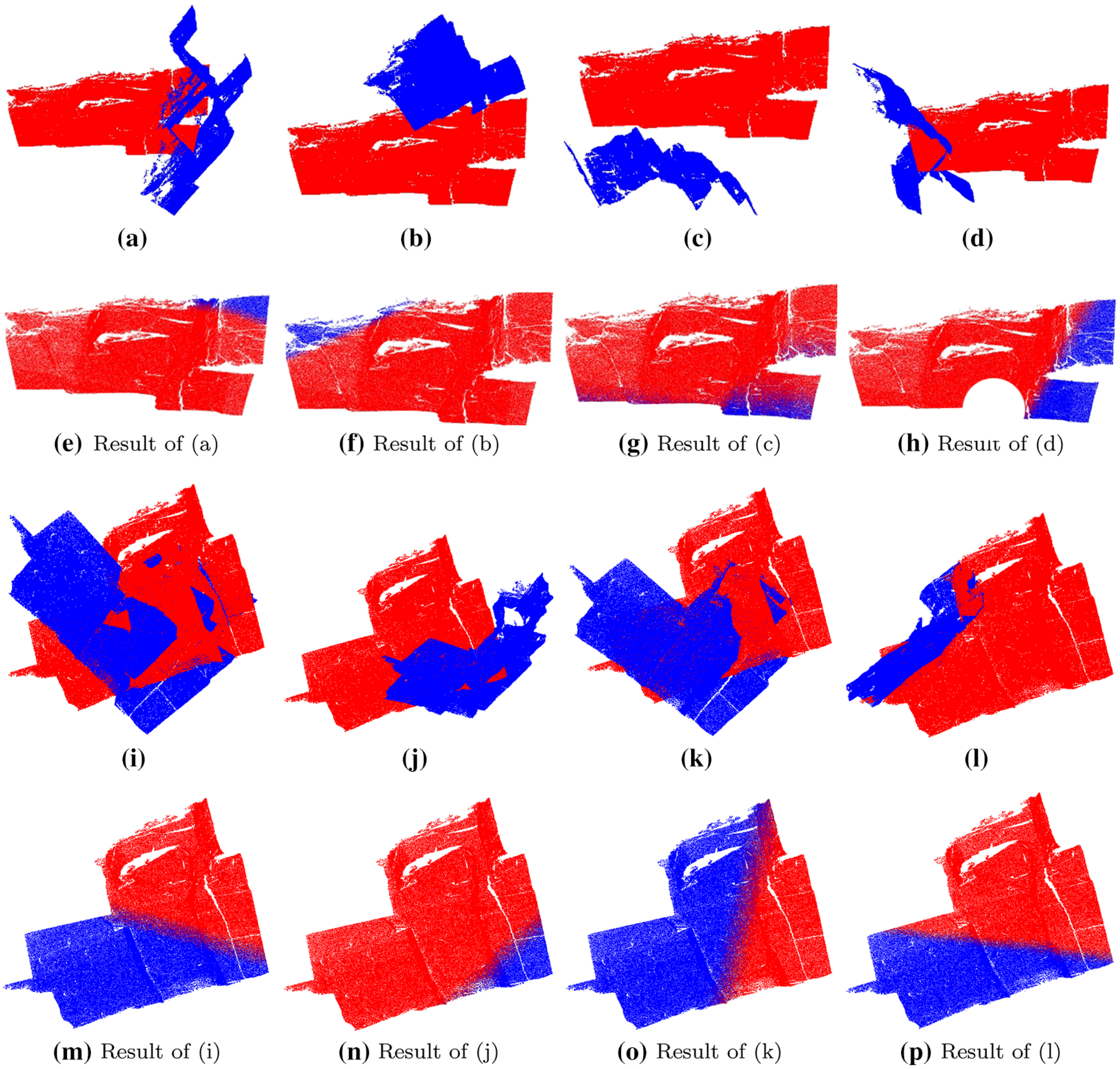


Fig. 10 Results achieved by using the proposed method with different initial position. **a**–**d** data 1 with different initial positions; **e**–**h** registration results of **a**–**d** using the proposed method. **i**–**l** data 2 with different initial positions; **m**–**p** registration results of **i**–**l** using the proposed method

Table 1 Results achieved by using the proposed method with different kinds of Gaussian noises and different initial positions

Different Gaussian noise		Different initial position	
δ	RMS error (m)	Position	RMS error (m)
0.00 Fig. 9a	0.0051	Fig. 10a	0.0052
0.10 Fig. 9b	0.0271	Fig. 10b	0.0051
0.15 Fig. 9c	0.0312	Fig. 10c	0.0051
0.20 Fig. 9d	0.0337	Fig. 10d	0.0050
0.00 Fig. 9i	0.0051	Fig. 10i	0.0051
0.10 Fig. 9j	0.0338	Fig. 10j	0.0051
0.15 Fig. 9k	0.0378	Fig. 10k	0.0052
0.20 Fig. 9l	0.0410	Fig. 10l	0.0051

the initial position has low influence on performance of our method. The RMS error is shown in Table 1, and the results show that the proposed method has reliable performance for registration between rock mass point clouds, which contain different noises.

4.2 Comparison

4.2.1 Plane detection

Plane detection is an important part of this paper, and the performance of our method highly depends on the quality of planes in rock mass point clouds. Therefore, we first test the performance of the plane detection. To validate the performance of our method, we compared it with three different plane detection methods. The first method (method I) is the optimized RANSAC technique of Schnabel et al. [26]. The performance of RANSAC was improved by the localized

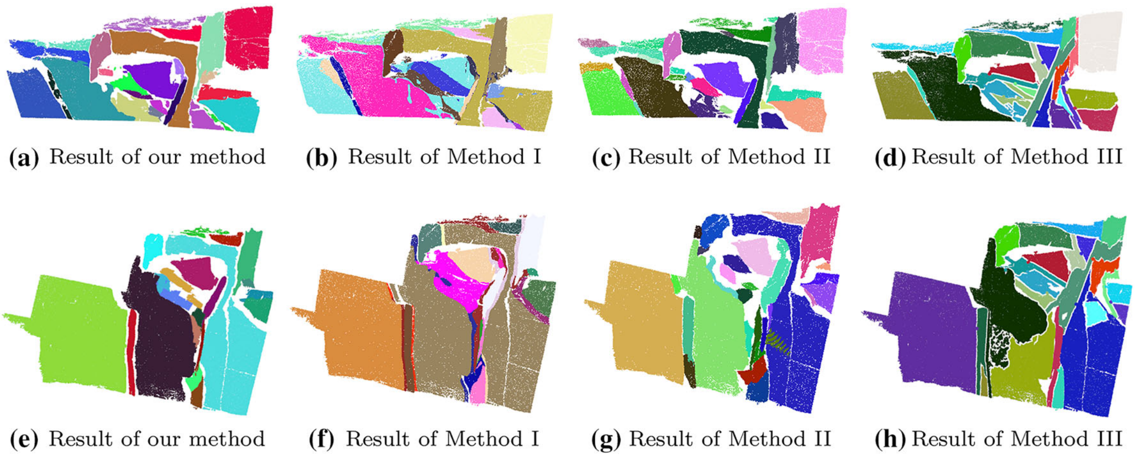


Fig. 11 Results of different plane detection methods on rock mass data 1 and data 2. Different colors represent different planes

Table 2 Accuracy and running time of different plane detection methods

Method	Data1			Data2		
	Number of planes	Accuracy (%)	Time (s)	Number of planes	Accuracy (%)	Time (s)
Our method	23	85.9	0.892	22	88.3	1.132
Method I	25	72.3	1.783	27	79.1	2.101
Method II	23	84.4	53.24	31	84.4	73.24
Method III	28	56.3	48.76	24	66.5	69.8

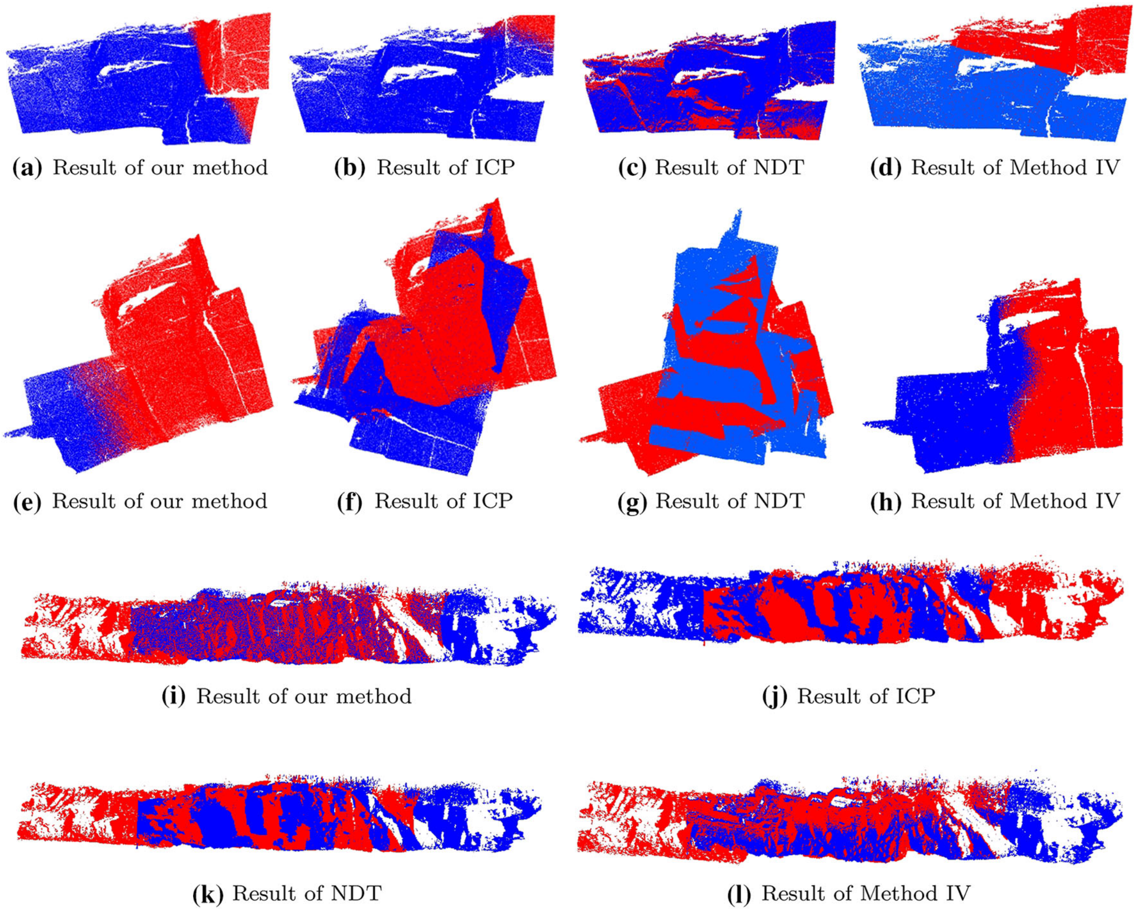
sampling strategy and was widely used in some open-source and commercial softwares. The second method (method II) [36] is an combination methods of HT and RG, which was applied to rock mass point clouds for plane detection. The third method (method III) is a simple RG method in open-source PCL. Data 1 (17 planes) and Data 2 (20 planes) are used for performance comparisons.

In order to find correct corresponding planes in the next procedure, the boundary of planes should be precise. Therefore, the detected planes should have clear boundary and do not contain any other planes nor intersect with them. The results of different method are shown in Fig. 11 and Table 2. The experimental results show that RANSAC can detect large-scale planes efficiently, but some mistakes are made at the boundary and the rough regions, shown in Fig. 11b, f. A completed plane is segmented into two or more planes using the simple RG method because of the different roughness and orientation in rock mass point cloud, shown in Fig. 11d, h. Therefore, a single method is not sufficient for plane detection from rock mass point clouds. As shown in Fig. 11c, g, method II takes advantages of both HT and RG methods which has a good performance on detecting planes in rock mass point cloud, but the high computation cost in the procedure of HT influences the efficiency of this algorithm. Compared with other methods, the proposed method can detect complete planes from rock mass point clouds efficiently and accurately. The planes detected by our method have clear boundary and can be used for polygon matching.

4.2.2 Registration

For performance comparisons, we used ICP and 3D normal distribution transform (3D-NDT) implementation for registration in point clouds, and these methods are available in the PCL [37]. What is more, a fast registration algorithm [19] based on spherical projection and feature extraction (method IV) is also used for comparisons. Both RMS error and running time of each method are computed for performance evaluation.

The result of each method is shown in Fig. 12. As shown in Fig. 12f, j, ICP is a local optimization and converges to a local minimum on data 2. With a good initial position, ICP can obtain an accurate transformation shown in Fig. 12b. Thus, ICP cannot be directly used for rocky points. As shown in Table 3, because of the procedure of coarse registration, the proposed method can obtain an accurate initial position, which can greatly improve the performance of our method. The method of 3D-NDT can improve the efficiency compared with ICP in some scenes, but it still has many difficulties in dealing with rock mass point clouds as shown in Fig. 12g. Method IV has a better performance than the method of ICP and 3D-NDT. This method converts point clouds to images based on spherical projection. SIFT is used to find corresponding relationship on images. Thus, color information is needed in this method. However, in some complex scenes, it is difficult to obtain accurate color information. In addition, at the procedure of spherical projection, some useful infor-


(a) Result of our method (b) Result of ICP (c) Result of NDT (d) Result of Method IV

(e) Result of our method (f) Result of ICP (g) Result of NDT (h) Result of Method IV

(i) Result of our method (j) Result of ICP

(k) Result of NDT (l) Result of Method IV

Fig. 12 Registration result of different methods on different rock mass data

Table 3 REM error and running time of different registration methods

Method	REM error (m)			Running time (s)		
	Data1	Data2	Data3	Data1	Data2	Data3
Proposed method	0.005	0.005	0.016	3.7	4.8	8.2
ICP	0.005	2.370	0.192	31.8	201.7	128.4
NDT	0.035	1.430	0.204	321.6	78.6	63.1
Method IV	0.017	0.005	0.037	27.5	21.3	33.5

mation may be removed. The RMS and running error are shown in Table 3. The experimental results show that simple transformation can be computed easily by all the methods. The proposed method has the best performance for registration in dealing with rough rock mass point clouds such as data 3. In summary, compared with other algorithms, the proposed method can be applied to registration between rock mass point cloud efficiently and accurately.

5 Conclusions and future work

This paper focuses on the field of 3D registration between rock mass point clouds. In rock engineering, planes detected from rock mass point clouds are one of the most important features, which can be used for the analysis of rock stability. Thus, a novel automatic registration method based on plane detection and polygon matching is introduced. A triple-region growing method is applied to detect planes on rock mass point clouds. This method is optimized for rock mass scenes and can be directly used on complex rock mass point clouds which contain irregular planes and different degrees of roughness. As plane features are relatively stable features in rock mass point clouds, the proposed method is robust to noise. Thus, compared with the other registration methods, our method is more efficient and accurate in dealing with rock mass point clouds. What is more, planes detected from rock mass point clouds can provide the initial primi-

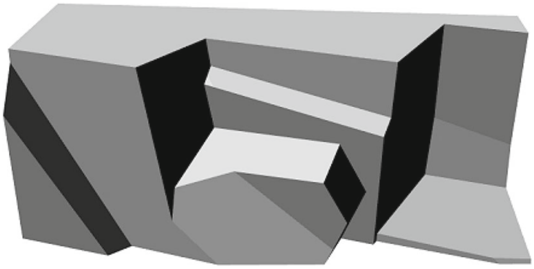


Fig. 13 Rock mass model reconstructed from data 1. This reconstruction method is based on our registration method, and the result of plane detection as shown in Fig. 5d

tives (planes) for 3D reconstruction of rock mass as shown in Fig. 13. However, the proposed method is designed for rock mass scenes and the irregular plane shapes are important criterion to find the correspondences between point clouds. The proposed method may have low performance in other scenes.

Acknowledgements This work is supported by the National Natural Science Foundation of China (No. 61471338), Youth Innovation Promotion Association CAS (2015361), Key Research Program of Frontier Sciences CAS (QYZDY-SSW-SYS004), Beijing Nova Program (Z171100001117048), Beijing Science and Technology Project (Z181100003818019) and President Fund of UCAS.

Compliance with ethical standards

Conflict of interest The authors certify that there is no conflict of interest with any organization for the present work.

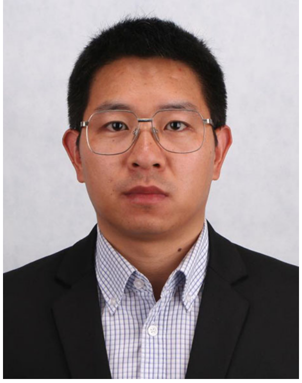
References

1. Yan, W.Y., Shaker, A., El-Ashmawy, N.: Urban land cover classification using airborne lidar data: a review. Remote Sens. Environ. **158**, 295–310 (2015)
2. Abellán, A., Oppikofer, T., Jaboyedoff, M., Rosser, N.J., Lim, M., Lato, M.J.: Terrestrial laser scanning of rock slope instabilities. Earth Surf. Process. Landf. **39**(1), 80–97 (2014)
3. Lin, H., Jizhou Gao, Y., Zhou, G.L., Ye, M., Zhang, C., Liu, L., Yang, R.: Semantic decomposition and reconstruction of residential scenes from lidar data. ACM TOG **32**(4), 66 (2013)
4. Mohammed, B., Ferdous, S.: 3d object recognition in cluttered scenes with local surface features: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **36**(11), 2270–87 (2014)
5. Xiao, J., Zhang, J., Adler, B., Zhang, H., Zhang, J.: Three-dimensional point cloud plane segmentation in both structured and unstructured environments. Robot. Auton. Syst. **61**(12), 1641–1652 (2013)
6. Montuori, A., Luzi, G., Stramondo, S., Casula, G., Bignami, C., Bonali, E., Bianchi, M.G., Crosetto, M.: Combined use of ground-based systems for cultural heritage conservation monitoring. In Geoscience and Remote Sensing Symposium, pp. 4086–4089 (2014)
7. Diez, Y., Roure, F., Lladó, X., Salvi, J.: A qualitative review on 3d coarse registration methods. ACM CSUR **47**(3), 45 (2015)
8. Vöge, M., Lato, M.J., Diederichs, M.S.: Automated rockmass discontinuity mapping from 3-dimensional surface data. Eng. Geol. **164**, 155–162 (2013)
9. Lato, M.J., Vöge, M.: Automated mapping of rock discontinuities in 3d lidar and photogrammetry models. Int. J. Rock Mech. Min. Sci. **54**, 150–158 (2012)
10. Tam, G.K.L., Cheng, Z.-Q., Lai, Y.-K., Langbein, F.C., Liu, Y., Marshall, D., Martin, R.R., Sun, X.-F., Rosin, P.L.: Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. IEEE Trans. Vis. Comput. Graph. **19**(7), 1199–1217 (2013)
11. Priest, S.D.: Discontinuity Analysis for Rock Engineering. Springer, Berlin (2012)
12. Restrepo, M.I., Ulusoy, A.O., Mundy, J.L.: Evaluation of feature-based 3-d registration of probabilistic volumetric scenes. ISPRS J. Photogramm. Remote Sens. **98**, 1–18 (2014)
13. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: IEEE International Conference on Robotics and Automation, 2009. ICRA'09. Citeseer, pp. 3212–3217 (2009)
14. Yang, M.Y., Cao, Y., McDonald, J.: Fusion of camera images and laser scans for wide baseline 3d scene alignment in urban environments. ISPRS J. Photogramm. Remote Sens. **66**(6), S52–S61 (2011)
15. Guo, Y., Sohel, F.A., Bennamoun, M., Wan, J., Lu, M.: Rops: a local feature descriptor for 3d rigid objects based on rotational projection statistics. In: 2013 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA). IEEE, pp. 1–6 (2013)
16. Mellado, N., Aiger, D., Mitra, N.J.: Super 4pcs fast global point-cloud registration via smart indexing. In Computer Graphics Forum, vol. 33. Wiley Online Library, pp. 205–215 (2014)
17. Mohamad, M., Ahmed, M.T., Rappaport, D., Greenspan, M.: Super generalized 4pcs for 3d registration. In: 2015 International Conference on 3D Vision (3DV). IEEE, pp. 598–606 (2015)
18. Yang, B., Zang, Y.: Automated registration of dense terrestrial laser-scanning point clouds using curves. ISPRS J. Photogramm. Remote Sens. **95**, 109–121 (2014)
19. Xian, Y., Xiao, J., Wang, Y.: A fast registration algorithm of rock point cloud based on spherical projection and feature extraction. Front. Comput. Sci. **13**(1), 170–182 (2019)
20. Elbaz, G., Avraham, T., Fischer, A.: 3d point cloud registration for localization using a deep neural network auto-encoder. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 2472–2481 (2017)
21. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In Sensor Fusion IV: Control Paradigms and Data Structures, vol. 1611, pp. 586–607. International Society for Optics and Photonics (1992)
22. Yang, J., Li, H., Jia, Y.: Go-icp: solving 3d registration efficiently and globally optimally. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1457–1464 (2013)
23. Pomerleau, F., Magnenat, S., Colas, F., Liu, M., Siegwart, R.: Tracking a depth camera: parameter exploration for fast icp. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3824–3829. IEEE (2011)
24. Borrmann, D., Elseberg, J., Lingemann, K., Nüchter, A.: The 3d hough transform for plane detection in point clouds: a review and a new accumulator design. 3D Res. **2**(2), 3 (2011)
25. Limberger, F.A., Oliveira, M.M.: Real-time detection of planar regions in unorganized point clouds. Pattern Recognit. **48**(6), 2043–2053 (2015)
26. Schnabel, R., Wahl, R., Klein, R.: Efficient ransac for point-cloud shape detection. In Computer graphics forum, vol. 26, pp. 214–226. Wiley Online Library (2007)

27. Oesau, S., Lafarge, F., Alliez, P.: Planar shape detection and regularization in tandem. In Computer Graphics Forum, vol. 35, pp. 203–215. Wiley Online Library (2016)
28. Poppinga, J., Vaskevicius, N., Birk, A., Pathak, K.: Fast plane detection and polygonalization in noisy 3d range images. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3378–3383 (2008)
29. Holz, D., Behnke, S.: Fast range image segmentation and smoothing using approximate surface reconstruction and region growing. In: International Conference on Intelligent Autonomous Systems, pp. 61–73 (2012)
30. Xiao, J., Zhang, J., Zhang, J., Zhang, H.: Fast plane detection for slam from noisy range images in both structured and unstructured environments, pp. 1768–1773 (2011)
31. Wang, J., Garratt, M., Anavatti, S.: Dominant plane detection using a rgb-d camera for autonomous navigation. In: International Conference on Automation, Robotics and Applications, pp. 456–460 (2015)
32. Gomes, R.K., de Oliveira, L.P.L., Gonzaga Jr., L., Tognoli, F.M.W., Veronez, M.R., de Souza, M.K.: An algorithm for automatic detection and orientation estimation of planar structures in lidar-scanned outcrops. Comput. Geosci. **90**, 170–178 (2016)
33. Oesau, S., Lafarge, F., Alliez, P.: Planar shape detection and regularization in tandem. Comput. Graph. Forum **35**(17), 2453–4 (2015)
34. Gigli, G., Mugnai, F., Leoni, L., Casagli, N.: Brief communication "analysis of deformations in historic urban areas using terrestrial laser scanning". Nat. Hazard. Earth Syst. Sci. **9**(6), 1759–1761 (2009)
35. Lato, M., Kemeny, J., Harrap, R.M., Bevan, G.: Rock bench: establishing a common repository and standards for assessing rockmass characteristics using lidar and photogrammetry. Comput. Geosci. **50**, 106–114 (2013)
36. Leng, X., Xiao, J., Wang, Y.: A multi-scale plane-detection method based on the hough transform and region growing. Photogramm. Rec. **31**(154), 166–192 (2016)
37. Holz, D., Ichim, A.E., Tombari, F., Rusu, R.B., Behnke, S.: Registration with the point cloud library: a modular framework for aligning in 3-d. IEEE Robot. Autom. Mag. **22**(4), 110–124 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Liang Hu is now a Ph.D. candidate in computer application technology at University of the Chinese Academy of Sciences. He received his bachelor's degree in software engineering from Xidian University in 2012. His research interests include computer graphic, point cloud processing and 3D reconstruction.



Jun Xiao is now a professor in University of the Chinese Academy of Sciences. He obtained his Ph.D.in the Graduate University of the Chinese Academy of Sciences in 2008. His research interests include computer graphics, computer vision, image processing and 3D reconstruction.

Ying Wang is now a professor in University of the Chinese Academy of Sciences. She received her Ph.D. in Beijing Institute of Technology in 1996. Her research interests include computer graphics, visualization and computer vision.