IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 19, 2020, accepted November 29, 2020, date of publication December 3, 2020,
date of current version December 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3042261

Developing a Reassembling Algorithm for Broken Objects

CAIQIN JIA[1], LIGANG HE[2], (Member, IEEE), XIAOWEN YANG[1], XINGCHENG HAN[3], BOBO CHANG[1], AND XIE HAN[1]

[1]School of Data Science and Technology, North University of China, Taiyuan 030051, China
[2]Department of Computer Science, University of Warwick, Coventry CV4 7AL, U.K.
[3]School of Information and Communication Engineering, North University of China, Taiyuan 030051, China

Corresponding author: Xie Han (hanxie@nuc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672473, in part by the Key Research and Development Program of Shanxi Province of China under Grant 201803D121081 and Grant 201903D121147, in part by the Natural Science Foundation of Shanxi Province of China under Grant 201901D111150, in part by the Science Foundation of North University of China under Grant XJJ201927, and in part by the Research Project supported by the Shanxi Scholarship Council of China under Grant 2020-113.

ABSTRACT The research on reassembling broken objects has many important applications, such as cultural relics restoration, medical surgery and solving puzzle. Because of the complicated surfaces of the fractured object pieces, it is not easy to extract salient features from them. It becomes even more difficult and very time-consuming to reassemble broken objects when the fragments are severely corroded or some of them are lost. In order to improve the accuracy and speed of 3D fragment reassembling, an effective and efficient fragment reassembling algorithm based on point clouds is proposed this article. This method first extracts keypoints and their concavity and convexity according to the symbolic projection distance of the point cloud, and then uses the local neighborhood information of the keypoints to construct a multi-scale covariance matrix descriptor. Furthermore, by calculating the similarity of the covariance matrix descriptors, the initial pairs of match points are obtained. Finally, the geometric constraints are gradually added to optimize the sampling so as to find good hypotheses as quickly as possible. By doing so, the search space is narrowed continuously in each iteration of the process to speed up the hypothesis test. We have conducted extensive experiments. The results show that the proposed method can fuse multiple features of the fragments effectively and achieve an outstanding matching effect on the defected fragments, and that the proposed method is faster than the existing methods in literature.

INDEX TERMS Object reassembly, symbolic projection distance, multi-scale covariance descriptor, geometric consistency, RANSAC.

I. INTRODUCTION

At present, the fragment reassembling has been of great application value in the fields of cultural relics restoration [1]–[3], medical research [4]–[7] and object recognition [8]. An important task in the process of fragment reassembling and repair is to recover the fragments based on the geometric information identifying the region of fracture of the adjacent fragments. Since the fragments are fragile and may be of large quantity, they can be cumbersome to handle, making it a problem to recompose the fragments by manual operations. With the development of computer and 3D scanning technology, the automatic fragment reassembling became available. However, the existing algorithms only work well with perfect fragments. They cannot be applied directly when there is the data loss on the fracture surface or the contour ambiguity caused by the long-term corrosion after the cultural relics are excavated. Therefore, the research on reassembling three-dimensional fragments has far-reaching theoretical and application values.

In the process of fragment reassembling, a 3D transformation matrix is determined by a certain method, so that the common sections of the fragments can be matched correctly. Depending on the thickness of the fragments, the fragment reassembling methods can be classified broadly into contour-based methods and fracture-based methods. However, due to the complicated and uneven shape of the fracture surface of the fragments, it is difficult to define the

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

significant features of the fracture surface during the reassembling process. Some fragments even have tiny fracture blocks on their surfaces due to corrosion, which results in the loss of data and makes the reassembling detection an even more challenging task. The contour-based method is mostly suitable for thinner fragments, while the method based on fracture surface is suitable for thicker fragments.

In order to make the fragment reassembling method more versatile, a 3D fragment reassembling algorithm based on keypoints is proposed in this article. This algorithm does not need to extract the contour lines or the fracture surfaces of the fragments, but focuses on the keypoints and their features of the neighborhood points for restoration.

The algorithm mainly consists of the steps of keypoint extraction, feature descriptors generation, feature matching and mismatching rejection. In particular, the multi-scale symbolic projection distance of point cloud is first calculated to detect keypoints and their concavity and convexity. The multi-scale Covariance Descriptor is then defined to describe the local surface features of the keypoints. Next, the initial matching of the fragments is obtained based on the similarity of the Covariance Descriptor descriptors.

The final step of this algorithm is mismatching rejection. Random Sample Consensus (RANSAC) [22] is one of the most widely used techniques for solving mismatching elimination. The RANSAC methods usually find the optimal solution using the brute-force search through an iterative evaluation. In each iteration, a minimal subset is sampled from the initial matching point pairs and used to estimate and validate a hypothesis. Such operations repeat until a satisfactory solution is obtained. However, although the RANSAC algorithm can estimate the correct parameters of the transformation matrix, the number of calculations will increase when there are very few inner points, which will slow down the process. Therefore, in this article a new effective sampling strategy is introduced to improve RANSAC, which gradually adds the geometric constraints to select the outlier-free subsets as quickly as possible and consequently generate more accurate hypotheses. The remaining matching point pairs can then be verified one by one to ensure more correct point pairs are recognized.

In the final step of our algorithm, the combination of the geometric consistency and the above improved RANSAC is used to eliminate false matching from the initial matching and achieve an excellent reassembling result.

The rest of this article is organized as follows. The related work is reviewed in Section 2. In Section 3, our fragment pairwise matching method is presented in detail. In Section 4, our improved RANSAC for eliminating mismatches is presented. The experimental results are presented and discussed in Section 5. Finally, the conclusions are drawn in Section 6.

II. RELATED WORK

A. PAIRWISE MATCHING

At present, many methods have been proposed in the field of 3D fragment reassembling, which can be roughly divided into the following two types: 1) fragment reassembling based on contour lines; 2) fragment reassembling based on fracture surface.

A fragment reassembling method based on contour lines obtains the contour lines of the fragment by boundary extraction, and determines whether it should be reassembled according to the matching degree of contour lines [9], [10]. The methods of this category are mostly used in the matching of 2D images, or the cases with very thinner fragment fracture surface, such as fresco matching [11], [12] and ceramics matching [13]. They are not applicable to the fragments with inconspicuous outlines and blurred boundaries.

The fragment reassembling method based on fracture surface is the most common method, which completes matching by identifying the similarities of the fractured regions between adjacent fragments. At present, the methods of this category are mostly based on the keypoints. Namely, the vertices with significant features are extracted according to the geometric information of each point and its neighbor points in the point cloud model. Several similar feature point pairs are obtained according to the similarity of the keypoints. A particular matching optimization algorithm is then used to determine whether the fractured surface matches.

The method proposed by Huang *et al.* [14] uses the integral invariant to extract the surface features of the fragments and realize the segmentation of the fracture surface, and then completes the fragment reassembly by surface features matching. Their method is difficult to identify and segment fracture surfaces for the small fragments. Pokrass *et al.* [15] proposed a partial shape matching method based on shape descriptors, which sampled the fracture surfaces to receive a number of sampling points and adopted the forward search method to obtain the optimal matching. Their method only finds similar parts and does not provide a point-to-point correspondence relationship.

Altantsetseg *et al.* [16] used a Fourier transform-based method to extract the curves on the fracture surfaces as the features to match the sections. Sánchez-Belenguer and Vendrell-Vidal [17] provided a general solution to the problem of unconstrained reorganization, in which the feature points of the fragments were first extracted to generate the PFH descriptors that were matched, and then the geometric constraint was used to perform mismatch elimination.

Dongjuan [18] proposed a virtual reassembling method for 3D bronze fragments based on the combination of principal component analysis and fast Fourier transform, in which the contour line of the model and the corner points on the contour line were extracted to divide the contour line into different curve segments, and then the fragments were reassembled according to the similarity of the segment curves. Chaohua [19] proposed a 3D cultural relic fragment reassembling method based on the thickness feature. It regarded the thickness information of the fragments as a sequence, achieved the coarse matching results by finding the longest common subsequence (LCS), and executed the iterative closest point (ICP) algorithm to complete fine matching.

Son *et al.* [20] proposed a surface signature descriptor to recompose the fragments. It performs the initial matching the fragments according to the spin images, and then conducts the mismatch elimination according to RANSAC and the distance from the point to the line. Wu and Wang [21] proposed a sand particle recombination algorithm based on fracture surface, which scanned the fracture surface and combined the segmentation effect of the corresponding two-dimensional image to recombine the fracture surface.

The key of the surface matching algorithms is the correct segmentation and feature matching of the fracture surfaces. However, these algorithms typically require the fragments to contain sufficient feature information. When the fragments are damaged severely on the fracture surface or the fragments have the very narrow fracture surface, it is difficult to ensure the correctness of the matching. Therefore, a reassembling method based on keypoints is proposed in this article, which improves the robustness of versatility of the method and is more accurate.

B. RANSAC

RANSAC is a robust estimation method introduced by Fischler and Bolles [22], which is divided into two processes: *Hypothesis* and *Test*. Its main idea is to use an iterative method to estimate the correct model from the data containing the outer points and then find the inner points.

In general, RANSAC is mostly used in image matching [23], [24]. When performing the homography matrix estimation in image matching, four corresponding point pairs are required to preserve the changes of the viewpoint, while three corresponding point pairs are required for the purpose of rotational and translational parameter estimation.

In this article, the RANSAC algorithm is used to filter the mismatched feature point pairs, which has to estimate the parameters of the transformation matrix as accurately as possible from the data samples containing incorrect matching points, and iteratively find the estimation result satisfying the error condition.

A classical RANSAC algorithm consists of the following steps: 1) Randomly select a minimum sample subset from the initial matched feature point pairs, which is three pairs of feature matching points, and set the upper limit of the number of iterations; 2) Compute the transformation matrix M based on RANSAC samples; 3) Determine whether the other feature point pairs in the sample set satisfy the transformation matrix M according to the error threshold; 4) If there are enough sample feature point pairs to satisfy M, then M is reserved, otherwise the estimation is considered wrong and the RANSAC sample is reselected; 5) Repeat steps 1-4 until the number of iterations reaches the upper limit.

The RANSAC has good noise resistance and can be better implemented. But it also has the disadvantages. Many scholars conducted the research on robust parameter estimation and proposed different improved RANSAC [25]. For instance, Lu *et al.* [26] adopted an improved RANSAC to eliminate mismatches, in which a pre-test is added based on the distance constraint, and then the samples were selected and verified according to the conventional RANSAC. Zhao *et al.* [27] added the distance and angle constraints to improve the sampling efficiency and estimated the initial matching set, and then refined the point set with a smaller threshold to obtain the optimal feature point matching set. Chu and Xi [28] designed the cascaded filter to remove false matches, which improves the proportion of inliers of initial matches and guarantees the accuracy of global motion parameter. In the existing literature, there is still no better trade-off between the evaluation reliability and speed. When evaluating the existing methods, each model must be evaluated for all the matches in order to obtain reliable evaluation results. Even if the current model has been proved to be optimal, the evaluation process cannot be terminated halfway. To solve this problem, this article uses the false matching rate as an evaluation criterion. When the false matching rate of a given model is worse than the current criteria value during the evaluation, there is no need to go through the rest of the evaluation any longer.

For most randomized methods, sampling results directly affect the runtime. The termination of the iterative algorithm relies largely on its ability to quickly identify the uncontaminated subsets. Therefore, to overcome the low efficiency of the RANSAC algorithm, this article identifies the minimum sample subset by gradually adding geometric constraints in *Hypothesis*, so that a more accurate sample subset can be extracted. In *Test*, the point pairs of the test set are tested one by one. Our method does not select the point pair from scratch in each RANSAC iteration, but reuses the previous calculation results, which speeds up the processing. Our experimental results show that that our algorithm can achieve better accuracy without compromising the efficiency.

III. PAIRWISE MATCHING BASED ON SALIENT CONCAVE-CONVEX REGIONS

Figure 1 illustrates a framework of the presented method. In this figure, the symbol projection distance is calculated to obtain the keypoint set of the source and target 3D model in *Step 1-source* and *Step 1-target*, respectively. In *Step 2-source* and *Step 2-target*, the descriptor sets of the source and target 3D models are generated respectively by computing the feature descriptor based on the keypoints and their neighborhood. In Step 3, the corresponding keypoint pairs are computed by measuring the similarity of feature descriptor between source model and target model; Finally in Step 4, the improved RANSAC is used to eliminate the mismatching pairs in the corresponding keypoint pairs to obtain the final keypoint matching pairs.

Step 1, Step 2 and Step 3 in Figure 1 are presented in detail in Section 3.1 and Section 3.2, respectively. Step 4 is will be presented in Section 4.

A. KEYPOINTS EXTRACTION

Due to the complicated geometrical structures on the surface of the cultural relics model, the point cloud data obtained

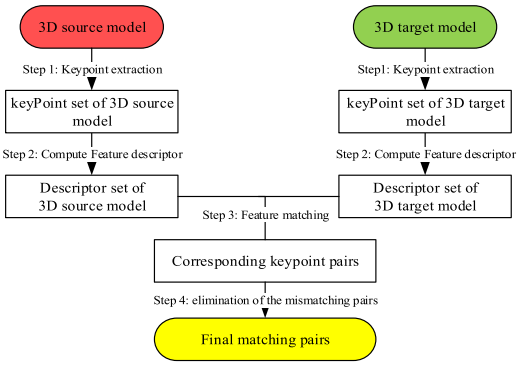


FIGURE 1. Frame Diagram of the method proposed in this article.

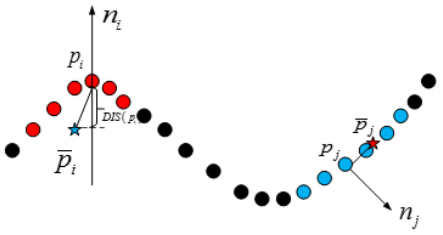


FIGURE 2. Point cloud symbolic projection distance.

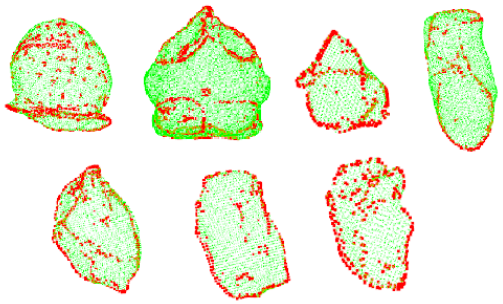


FIGURE 3. Results of keypoint detection.

by 3D scanners is very large. There may be a large amount of redundant data. In order to improve the quality of the surface reconstruction and reduce the influence of noise on the extraction of the geometric features, it is necessary to simplify the data of the point cloud model. In this article, the point cloud data are first simplified conformally by uniform sampling [29], that is, a three-dimensional voxel grid is created for the point cloud, and then the point closest to the center point of each voxel is taken as the sampling point. The keypoints are then determined by calculating the symbolic projection distance, and the concavity and convexity of keypoints are judged according to the symbol of the symbolic projection distances.

For any point p_i in the point cloud, the centroid in its spherical neighborhood is found to form a vector from point p_i to centroid of its neighboring point, where the projection distance is formed by projecting the vector onto the normal vector of point p_i. The smoother the surface is, the closer to zero the projection distance is, and vice versa.

For a given point cloud $\{p_1, p_2, \cdots, p_M\}$, when a query point p_i is given, a spherical neighborhood $N(p_i)$ is constructed by taking p_i as the center and r as the radius, and a covariance matrix is also constructed:

$$C_{p_i} = \sum_{j \in N(i)} \left(p_{ij} - \bar{p}_i\right)\left(p_{ij} - \bar{p}_i\right)^T \tag{1}$$

where $\bar{p}_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij}$, p_{ij} is the neighborhood points of point p_i, and N is the number of points in $N(p_i)$.

Covariance matrix C_{p_i} is a 3×3 matrix, for which the singular value decomposition (SVD) is executed to get:

$$[U, S, V] = svd\left(C_{p_i}\right)$$

in which, the feature vector corresponding to the minimum eigenvalue is the normal vector, that is, the third column of the matrix C_{p_i}. In order to prevent the ambiguity arising from the direction of the normal vector, it is necessary to standardize the normal vector.

The normal vector is standardized through the given point p_i and the neighborhood points $\left\{p_{i1}, \cdots, p_{ij}, \cdots, p_{iN}\right\}$, $j = 1, \cdots, N$

$$n_{p_i} = \begin{cases} n_{p_i} & \left\langle n_{p_i}, \overrightarrow{p_i - p_{ij}} \right\rangle \geq 0 \\ -n_{p_i} & \left\langle n_{p_i}, \overrightarrow{p_i - p_{ij}} \right\rangle < 0 \end{cases} \tag{2}$$

The multi-scale symbolic projection distance is defined:

$$\mathrm{SD}_r\left(p_i\right) = \left\langle \overrightarrow{\left(p_i - \overline{p_i}\right)}, n_{p_i} \right\rangle$$

$\langle a, b \rangle$ indicating the dot-product operation

In Figure 2, the red dot indicates the points in salient regions, the blue dot indicates a relatively flat point, and the five-pointed star indicates the centroid in the corresponding neighborhood. Procedure for calculating the symbolic projection distance is shown in Figure 2.

When the absolute value of SD is greater than a certain set threshold, the point is regarded as being of salience and can be considered as the keypoint in the next step in this article. Otherwise it is ignored. When SD is greater than 0, the point is a concave point in the neighborhood. On the contrary, when SD is a negative value, the point is a convex point. The extreme points of the SD in the local neighborhood are taken as keypoints. The concavity and convexity of each keypoint are also identified.

$$\begin{aligned} &\forall j\ SD_r\left(p_i\right) > SD_r\left(p_j\right), \left\|p_i - p_j\right\| \leq r \wedge SD_r\left(p_i\right) > 0 \\ &\forall j\ SD_r\left(p_i\right) < SD_r\left(p_j\right), \left\|p_i - p_j\right\| \leq r \wedge SD_r\left(p_i\right) < 0 \end{aligned} \tag{3}$$

The detection results of the keypoints for the model venus are shown in Figure 3. The green points are the original model points while the red points are the detected keypoints. It can be seen that there are fewer feature points in the smooth region, but more feature points in the salient region.

B. KEYPOINT MATCHING BASED ON SALIENT FEATURES

Usually, the corresponding regions of matching fragments must be complementarily concave and convex. Therefore, if the fragments are intact, the neighborhood of the corresponding fracture region should also be similar. Based on this attribute, a multi-scale covariance matrix descriptor is utilized to describe the initial matching of the model in this article.

1) MULTI-SCALE COVARIANCE DESCRIPTOR

Numerically, the covariance matrix can be used to represent the interaction of multiple variables. Covariance was originally used as a descriptor in image processing [30], and later in 3D point cloud registration [31]. Covariance matrix descriptors can fuse different types of features, reflecting not only the spatial and geometric properties of the described objects, but also the correlation of these features. We use the covariance matrix to illustrate the relationship among different features. The values distributed in the main diagonal of the covariance matrix indicate the variation for each feature. Other elements represent the similarity between the two characters. According to the definition of covariance matrix descriptor, a set of random variables correspond to a set of geometric features extracted from a point cloud, such as 3D coordinates, normal vectors, curvature, and so on. Therefore, it is necessary to first construct the feature selection function. In the function, a keypoint p is given, and p_i is the point in the point cloud that can satisfy all the spherical neighborhoods within the radius r of the distance point p(denoted by $N(p)$).

$$\begin{aligned} \Phi(p) &= \left\{ \varphi_{p_i} \middle| \, |p - p_i| \le r \right\} \\ \varphi_{p_i} &= \{\cos\alpha, \cos\beta, \cos\gamma, \delta_1(p), \delta_2(p), \delta_3(p), H(p)\} \end{aligned} \tag{4}$$

As shown in Figure 4, α represents the angle between the normal vector of point p_i and the line segment $\overrightarrow{pp_i}$, β represents the angle between the normal vector of point p and the line segment $\overrightarrow{pp_i}$, and γ represents the angle between the point n_{p_i} and n_p. In order to reduce the calculation time, the cosine value is calculated when calculating the angle values. H represents the average curvature of the point p, obtained by the calculation method in [32]. All of the extracted features are invariant for the rigid transformation, which is of great importance for matching. These features can be calculated by:

$$\begin{aligned} \cos\alpha &= \frac{\overrightarrow{pp_i} \cdot n_{p_i}}{\|pp_i\|_2} \quad \delta_1(p) = \frac{\lambda_1 - \lambda_2}{\lambda_1} \\ \cos\beta &= \frac{\overrightarrow{pp_i} \cdot n_p}{\|pp_i\|_2} \quad \delta_2(p) = \frac{\lambda_2 - \lambda_3}{\lambda_1} \\ \cos\gamma &= n_{p_i} \cdot n_p \quad \delta_3(p) = \frac{\lambda_3}{\lambda_1} \end{aligned} \tag{5}$$

where $\lambda_1, \lambda_2, \lambda_3 (\lambda_1 \ge \lambda_2 \ge \lambda_3)$ indicate the eigenvalue obtained by performing the principal component analysis (PCA) calculation based on the neighborhood of the point p. $\lambda_i (i = 1, 2, 3)$ describes the range of data distribution on the corresponding eigenvector. The feature $\delta_1(p)$, $\delta_2(p)$ and $\delta_3(p)$ describe the structural features of the neighborhood of point p. If $\lambda_1 \ge \lambda_2$ and $\lambda_1 \ge \lambda_3$, $\delta_1(p)$ is greater than $\delta_2(p)$ and $\delta_3(p)$. In this case, a linear structure is represented by the neighborhood of point p. If $\lambda_1 \approx \lambda_2 \ge \lambda_3$, $\delta_2(p)$ is the largest one. A planar structure is represented by the neighborhood of point p in this case. Finally, $\lambda_1 \approx \lambda_2 \approx \lambda_3$ means that $\delta_3(p)$ is larger than $\delta_1(p)$ and $\delta_2(p)$, which means that a scatter structure is represented by the neighborhood of point p.

Given a keypoint p, its covariance descriptor can be expressed as:

$$C_r(\varphi(p, r)) = \frac{1}{N-1} \sum_{i=1}^{N} (\varphi(p_i) - \mu)(\varphi(p_i) - \mu)^T \tag{6}$$

where μ is the average of a series of eigenvectors $\{\phi_{p_i}\}$, and N is the number of points in $N(p)$. The result is a $7*7$ positive definite matrix C_r, whose diagonal elements represent the change in the distribution of each feature while the elements on the non-diagonal represent the correlation between the feature point and the neighborhood point. According to the geometric characteristics of a point cloud, the use of covariance descriptors can guarantee the rotation and translation invariance and viewpoint invariance of the model.

2) SIMILARITY OF MULTI-SCALE COVARIANCE DESCRIPTORS

In this article, we use the metric introduced by Förstner and Boudewijn [33]. It is defined in Equation (7) and widely used in the comparison of the covariance descriptors in Riemann Metric.

$$\delta(X, Y) = \sqrt{trace\left(\log^2\left(X^{-(1/2)} Y X^{-(1/2)}\right)\right)} \tag{7}$$

where X and Y represent two covariance matrices, respectively, and $\log(\cdot)$ represents the logarithm of the matrix. This metric incurs a large amount of calculation. In recent years, researchers proposed to use the logarithmic European Riemannian Metric instead [34], which is defined in Eq. 8.

$$\delta(X, Y) = \|\log(X) - \log(Y)\|_F \tag{8}$$

where $\|\cdot\|_F$ represents the F norm of the matrix. The smaller the value, the smaller the difference between two models.

In order to improve the description to the keypoints, our method obtains the multi-scale Covariance Descriptor, which

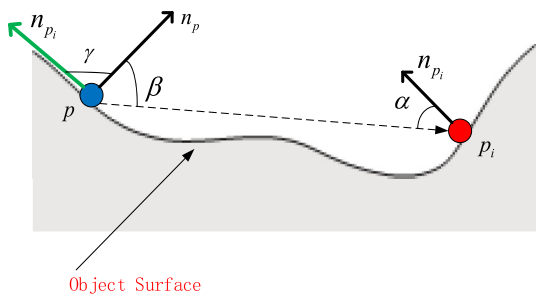


FIGURE 4. Local geometric features of the point cloud model.

changes the number of neighborhood points by changing the radius r. The covariance descriptors at different scales can be described as:

$$C_M(p) = \{C_r(\varphi(p, r)), \forall r \in \{r_1, \cdots, r_s\}\} \tag{9}$$

The distance between multi-scale Covariance Descriptors can be measured by the mean value of the similarity between sets of single-scale covariance descriptors, as shown in Equation (10).

$$\delta_M\left(C_M^1, C_M^2\right) = \frac{1}{s}\sum_{i=r_1\cdots r_s}\delta\left(C_i^1, C_i^2\right) \tag{10}$$

where C_i^1 and C_i^2 represent the covariance descriptors the radius at the scale of $r_i\,(i = 1, \cdots, s)$ respectively, and $\delta\left(C_i^1, C_i^2\right)$ represents the similarity of the covariance descriptors at the single scale calculated by Equation (8).

IV. ELIMINATING MISMATCHING THROUGH THE IMPROVED RANSAC

There are commonly used methods for eliminating the mismatching, such as Geometric Consistency [35], the game theory [36] and RANSAC [37]. The procedure of a game theoretic method converges to a limited set of correspondences by successively removing false correspondences. The Geometric Consistency method divides the potential correspondences into different groups using the geometric constraints, and selects the group with the largest geometrical consistency to estimate the transformation matrix. The RANSAC method usually finds the optimal solution using the brute-force search through an iterative evaluation. At each iteration, a subset of candidate correspondences are randomly selected. However, in order to reduce the number of iterations and increase the effectiveness of RANSAC as much as possible, purifying the feature matching point pairs is an indispensable step. The accuracy of the randomly selected samples also directly affects the number of iterations of the RANSAC algorithm. Therefore, in this article, the minimum subset of RANSAC is determined by gradually adding geometric constraints. In each RANSAC iteration, the search space is reduced continuously to increase the efficiency of the algorithm, so as to speedup the RANSAC algorithm and improve its accuracy.

A. SAMPLING

Complementary fragments have similar structures in the corresponding regions. According to this property, there must be similar geometric feature structures between the correct feature matching point pairs. As a matter of fact, the uncontaminated samples should be selected as early as possible. It is because this strategy can contribute not only to obtaining the optimal parameters of the model, but also to speeding up the convergence of the algorithm. Therefore, in order to find a better sampling set from initial feature matching point pairs set, the minimum sampling subset (which is three pairs of feature matching point pairs) is determined by gradually adding geometric constraints to ensure the accuracy of the initial sample.

Assume that P and Q are the two fragments of point cloud models to be recomposed, respectively, S is the initial feature matching point pairs, $S = \{(p_1, q_1), (p_2, q_2), \cdots, (p_n, q_n)\}$, and n is the number of feature matching point pairs.

We select a pair of point pairs (p_i, q_i) randomly in the initial feature matching point pairs set S as the first fixed point pair, and determine the second feature matching point pair (p_j, q_j) in the remaining initial point pair set S according to the following three Rules, in which, $d(x, y)$ represents the Euclidean distance between point x and point y, $\varepsilon_d > 0$ is the distance threshold. $\varepsilon_{angle} > 0$ is the angle threshold, and $\varepsilon_r > 0$ is the direction threshold.

1) Distance consistency constraint:

$$\left|d\left(p_i, p_j\right) - d\left(q_i, q_j\right)\right| < \varepsilon_d,$$

2) Angle consistency constraint:

$$\left|\left\langle n_{p_i}, n_{p_j}\right\rangle - \left\langle n_{q_i}, n_{q_j}\right\rangle\right| < \varepsilon_{angle},$$

3) Direction constraint:

$$\left|\frac{\left\langle p_i - p_j, -n_{p_j}\right\rangle}{\left\|p_i - p_j\right\|} - \frac{\left\langle q_i - q_j, n_{q_j}\right\rangle}{\left\|q_i - q_j\right\|}\right| \le \varepsilon_r$$

Computing rigid transformation requires at least three pairs of point sets. Therefore, a pair of feature matching point pairs (p_k, q_k) need to be randomly selected from the remaining initial matching point pairs set S. To prevent the randomly selected third point pairs from being collinear with the previous two pairs of points, the selection of the third point pairs must satisfy formula (11). The initial rigid transformation matrix $T_0 = (R_0, t_0)$ can be calculated, and the alignment error ε_0 can be calculated based on root mean square error of matching point pairs.

$$\left|\left(p_i - p_j\right) \times \left(p_i - p_k\right) - \left(q_i - q_j\right) \times \left(q_i - q_k\right)\right| < \varepsilon_p \tag{11}$$

where $\varepsilon_p > 0$ is a constant.

B. HYPOTHESIS EVALUATION

The three pairs of point pairs identified in the Section 4.1 are used as the candidate set. The remaining initial feature matching point pairs are used as the test set. An alternative method of evaluating the initial estimation for the transformation is to add each point pair in the test set to the candidate set one by one and re-compute the transformation which is outlined in Algorithm 1. This evaluation method is indeed slower since each point pair in the test set needs to be computed separately. But it is more accurate.

This article proposes three improvements over the traditional RANSAC algorithm. First, we set the optimal matching ratio. Namely, the correct matching ratio of the point pair is calculated in each iteration of the algorithm. Assume that the optimal matching ratio is 80%. If the incorrect matching ratio reaches 20%, it is impossible to achieve the optimal result. So the iteration can be ended early. The search space continues to decrease as the number of iterations increases. Therefore, the algorithm will converge faster.

Algorithm 1 New Hypothesis Evaluation

Input: initial transformation estimation (R_0, t_0), error ε_0, initial point pairs in sample sets (U_s, V_s), s = 1,2,...m, test sets (U_t, V_t), t = 1,2,...,n
Output: inlier point pairs

Begin
  for each corresponding points (u_t, v_t) in (U_t, V_t) **do**
    Calculate (R_i, t_i) that aligns the points in $V_s \cup v_t$ to the points in $U_s \cup u_t$
    Calculate the corresponding alignment residual error ε_i
    if $\varepsilon_i - \varepsilon_0 < threshold$
      (u_t, v_t) is considered as an inlier
    end if
  end for
End

Second, when estimating the rotation parameters in each RANSAC iteration, our method only adds a pair of feature matching point pairs for verification based on the minimal subset, rather than increase the number of pairs as the matching consistent set increases (i.e., the number of correct matching point pairs increases as the number of iterations increases, and consequently it becomes more and more complicated to estimate the rotation parameters). To reduce the computational cost of the RANSAC algorithm, our method places the correct matching point pairs in the queue according to the first-in-first-out (FIFO) principle, ensuring that only four pairs of feature matching point need to be calculated each time (the size of minimum subset is 3).

Third, the correct point pairs that have been calculated will not be calculated in subsequent iterations. The feature point pairs that are deemed correct in each iteration are taken as a class. Each type of point pair corresponds to a set of transformation. Therefore, our method does not need to select the point pair from scratch in each RANSAC iteration, but reuses the previous calculation results, which accelerates the hypothetical evaluation process.

The mismatching elimination algorithm based on the improved RANSAC proposed in this article is shown in Algorithm 2. This method first sets the maximum number of iterations *iter_max* and the mismatching ratio of feature matching point pairs η. It starts with extracting the minimum sampling subsets (U_s, V_s) from (U, V) according to the sampling method proposed in this article, in which each element in U_s has a corresponding element in V_s. Then the remaining element pairs constitute the test sets (U_t, V_t), where $U_t = U \backslash U_s$ and $v_d \in V_t = V \backslash V_s$. This method verifies the point pairs in the set of remaining initial point pairs one by one to increase the accuracy of the algorithm. For each additional point pair, a new transformation matrix $T_i = (R_i, t_i)$ and alignment error e_i are calculated. If $e_i - e_0$ less than a user-defined threshold, the point pair is added to the potential consistent set C. When verifying whether the added point pair

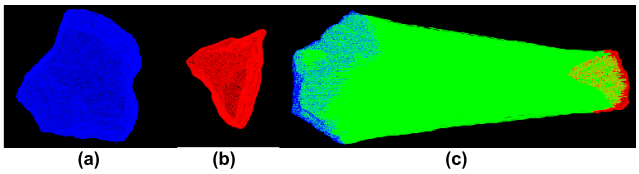


FIGURE 5. Initial matching of the brick model; (a) and (b) are original brick fragments, and (c) is initial matching between the two models.

meets the conditions of the consistent set, the error matching ratio of the point pair is calculated. If the error matching ratio exceeds η, this algorithm re-selects the minimum sampling subset for evaluation until the correct point pair matching ratio is met or the algorithm reaches the maximum number of iterations.

Algorithm 2 Eliminating Mismatching Based on the Improved RANSAC

Input: corresponding 3D point sets pairs (U, V)
Output: $T = (R, t)$ rotation R and translation t

Begin:
set Mismatching rate η
$iter \leftarrow 0$
$C \leftarrow 0$
$Countmismatch \leftarrow 0$
while $iter < iter\_\max$ **do**
$(U_s, V_s) \leftarrow getMinimalSampleSet(U, V)$
$e_s \leftarrow rmse\,(U_s, V_s)$
add (U_s, V_s) *to the potential Queen C*
  if $Countmismatch/n < \eta$
    randomly select $u_d \in U_t$ *and* $v_d \in V_t$
    $e_d \leftarrow rmse\,((u_d, v_d), C)$
    if $e_s - e_d < threshold$ *then*
      add u_d, v_d *to the potential Queen C*
      DelQueen (C)
    else $Countmismatch \leftarrow Countmismatch + 1$
    end if
  end if
end while
End

V. EXPERIMENTS

The experimental data in this article are obtained from the fragment data published by Vienna University of Technology [38]. All the experiments have been conducted on a computer with 3.4 GHz CPU and 16 GB of memory. The algorithm proposed in this article is applied to different fragments in the public data set to evaluate its effectiveness. A set of fragment models in the public data set are shown in Figure 6, in which (a) is the brick models, (b) is the venus models, (c) is the cake models, (d) is the sculpture models, and (e) is the gargoyle models.

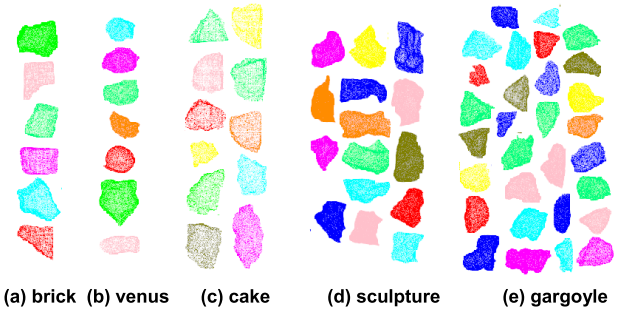


FIGURE 6. Original models.

TABLE 1. Fragment introduction.

#name	brick	cake	gargoyle	sculpture	venus
#number	6	11	30	15	7

In this article, the experiments are carried out on multiple point cloud models in the dataset. TABLE 1 show all the fragments used in the experiments, where the row #name is the name of the fragment and the row #number is the number of fragments contained in each model.

A. RESULTS OF COVARIANCE DESCRIPTOR

The 3D covariance descriptor has only one parameter: the combined feature vector. The selection of the features used for the construction of the covariance descriptor is a critical step to improve the performance of 3D feature matching. This is because the improper features will have the negative in?uence on the matching rate. Thus, this article evaluates the performance of covariance descriptors with the combination of different features.

Given a source model, target model and a ground-truth transformation, the keypoints of the source and the target model are extracted by our method and are described as the source model and target model features by the covariance matrix descriptor. The similarity between each feature descriptor of the source model and those of all target models is calculated based on formula (8). If the distance is less than a threshold, the source model feature is considered to match the target model feature. Further, the target model is transformed according to the ground-truth, and the Euclidean distance of the corresponding point pair between the target model and the corresponding source model is calculated. If the distance is small enough, the point pair is considered a correct positive, which is defined as N_{CP}. In addition, the number of source model features is defined as the total number of positives, which is defined as N_{TP}.Using this information, the correct matching rate can be computed as:

$$correct\ matching\ ratio = \frac{N_{CP}}{N_{TP}}$$

We tested the quality of the descriptor proposed in this article by adding different levels of Gaussian noise to the target model, where the noise deviation was set in the range of [0.1,0.5]. The curve can be generated for our experiments by varying the noise deviation. The result of the combination of different features is presented in Formula 12. Among these feature vectors, F_3 is our proposed feature vector. The features in F_1, F_2 and F_3 are all relatively calculated based on the relative relationship between the keypoint p and its neighboring p_i, while some features in F_4, F_5 are absolutely calculated based on the absolute position of the neighboring points around the keypoint. For example, $x_{p_i}, y_{p_i}, z_{p_i}$ are the absolute position of a neighboring point of keypoint p, $n_{x_{p_i}}, n_{y_{p_i}}, n_{z_{p_i}}$ are the normal of a neighboring point of keypoint p.

$$\begin{aligned}
F_1 &= (\cos\alpha, \cos\beta, \cos\gamma) \\
F_2 &= (\delta_1(p), \delta_2(p), \delta_3(p), H(p)) \\
F_3 &= (\cos\alpha, \cos\beta, \cos\gamma, \delta_1(p), \delta_2(p), \delta_3(p), H(p)) \\
F_4 &= (x_{p_i}, y_{p_i}, z_{p_i}, \cos\alpha, \cos\beta, \cos\gamma, \delta_1(p), \delta_2(p), \\
&\quad \delta_3(p), H(p)) \\
F_5 &= (x_{p_i}, y_{p_i}, z_{p_i}, n_{x_{p_i}}, n_{y_{p_i}}, n_{z_{p_i}}, \cos\alpha, \cos\beta, \cos\gamma, \\
&\quad \delta_1(p), \delta_2(p), \delta_3(p), H(p))
\end{aligned} \tag{12}$$

We use different models including brick, cake, venus, gargoyle and sculpture models to compute the average correct matching rate of the covariance descriptor, and the results are shown in Figure 7. It can be seen that the performance of the covariance descriptor formed by F_1, F_2 and F_3 is better than that of the covariance descriptor formed by F_4, F_5.Moreover, the correct matching rate of the covariance descriptor formed by decreases rapidly with the increase of noise. The reason is because some of the features in F_4, F_5 are not invariant to rigid transformation and are sensitive to noise, resulting in a decrease in the description ability of the covariance descriptor. At the same time, the comparison results of F_1 and F_2 in Figure 7 show that the change of the eigenvalue and surface variance are more stable than the change of angle. In addition, it can be observed from Figure 7 that as the number of features with noise resistance and rigid transformation in the feature vector F_i $(i = 1, 2, 3)$ increases, the performance of the covariance descriptor formed by the feature vector will also be improved. These results show our proposed feature vector has good robustness to noise.

B. RESULTS OF PAIRWISE MATCHING

Figure 8 shows the results of pairwise matching experiments for multiple fragment models. As shown in Figure 8, multiple fragment models match each other completely. It can be seen from Figure 8(5) that even if some models are missing, correct matching results are still obtained.

Figure 9 shows the pairwise matching results of the brick fragments. We tried to match any two pieces of the brick models. Our algorithm found 8 pairs of matching results, in which the fragment pairs 1-2, 1-3, 4-5, 4-6 and 5-6 are full match, 1-4, 2-4 and 2-5 are partial match. All these reassembly results are correct. These results suggest that our

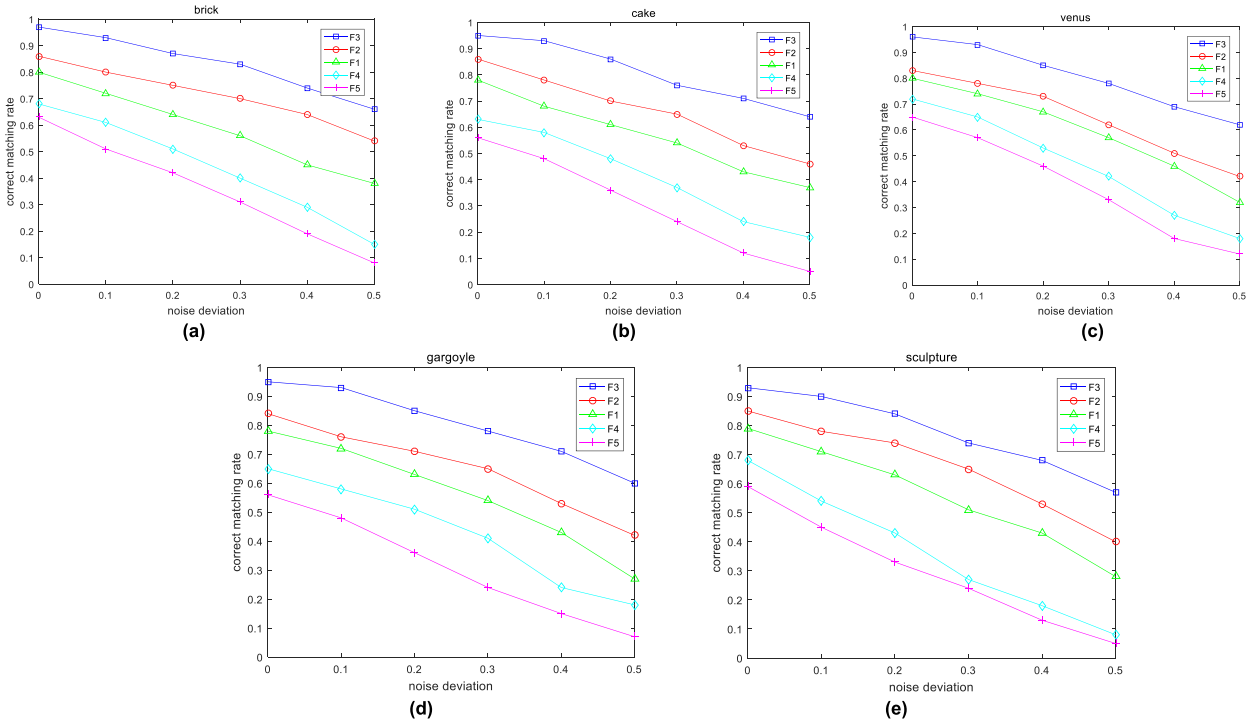


FIGURE 7. Performance of covariance matrix descriptor formed by different feature vector.

TABLE 2. Execution times (in seconds) of the algorithms and the breakdown of the excution times spent in finding the matches for the brick fragments.

Brick	#P		#F		T_{init}	T_{mis}	T_{total}		
							[14]	[20]	Ours
1-2	48195	15409	7586	5497	8.29	3.24	20.26	15.62	11.53
1-3	48195	18820	7586	6410	9.03	3.58	23.48	17.38	12.61
1-4	48195	30466	7586	5560	11.05	4.18	28.29	21.46	15.23
2-4	15409	30466	5497	5560	8.03	3.11	22.34	18.37	11.14
2-5	15409	34632	5497	6390	9.65	3.62	20.27	14.82	13.27
4-5	30466	34632	5560	6390	10.96	4.05	27.73	20.76	15.01
4-6	30466	15555	5560	4651	7.84	3.06	19.86	13.24	10.9
5-6	34632	15555	6390	4651	9.83	3.93	24.62	17.45	13.76

algorithm can accurately handle not only full matching but also partial matching.

The detailed experimental records of brick pairwise matching are listed in TABLE 2, where #P is the total number of points in the models after uniform sampling, #F is the number of detected keypoints, T_{init} is the time spent in finding initial matching, T_{mis} is the time in eliminating mismatches, T_{total} is the total time spent in finding the matches. TABLE 2 also compares the time spent by our algorithm and the time by the search algorithms presented in [14] and [20]. It can be seen that our algorithm almost halves the time spent by the algorithm in [14]. This is because the method in [14] uses the whole point cloud to compute the final reconstruction, while our method uses a reduced set of keypoints. It can also be observed from the TABLE 2 that the time spent by our method is much less than that by the method in [20]. The reason is because the method in [20] converts the 3D point cloud model into the 2D spin image for description. In addition, different points in the space have the same coordinates after transformation, resulting in more errors when comparing the similarity of spin image. Also the original RANSAC method is used to eliminate mismatches in [20], which is another factor why the algorithm in [20] is slower than our algorithm.

C. RESULTS OF COMPARISON WITH EXISTING METHODS

In order to better evaluate the performance of the proposed algorithm, we compares our pairwise matching with the existing methods, including those in [18] and [19]. We used

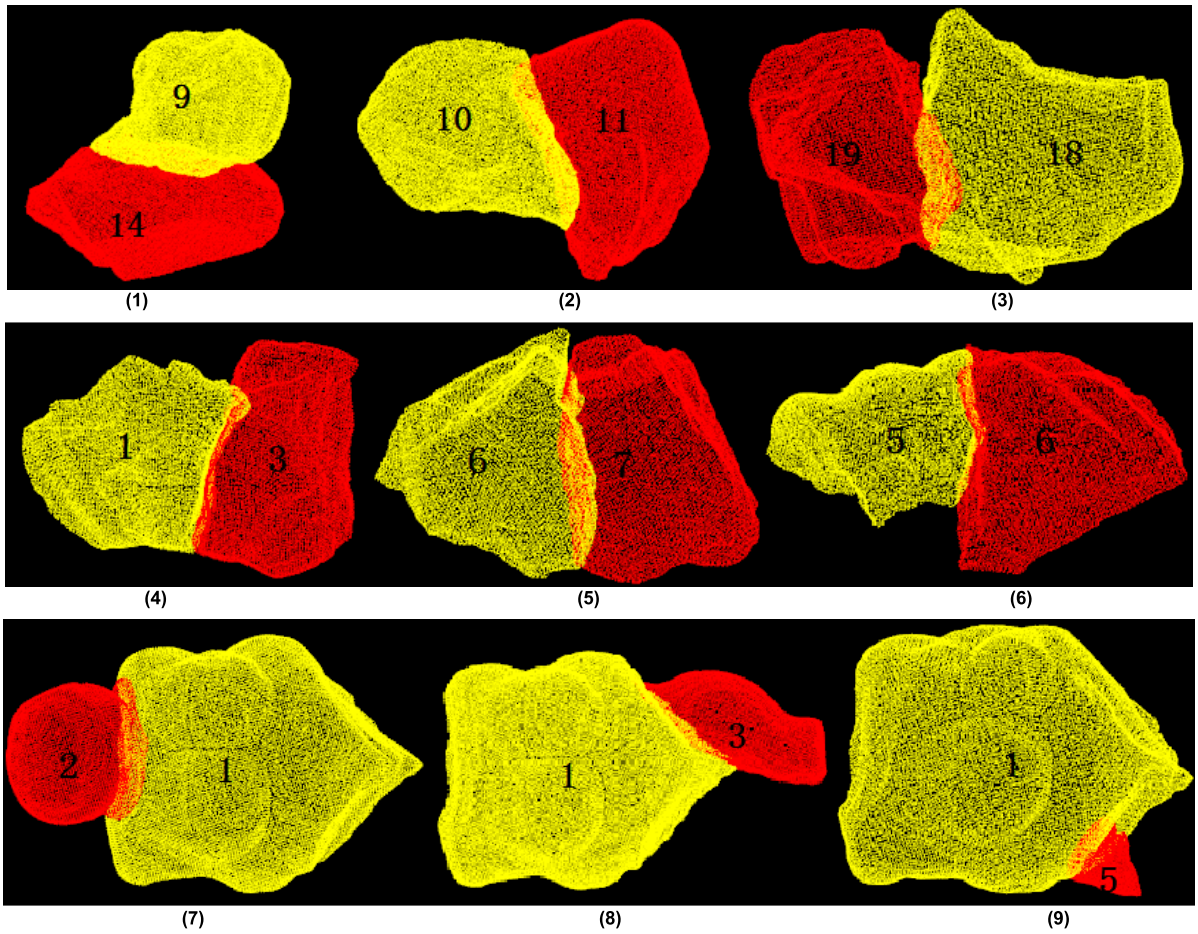


FIGURE 8. Pairwise matching results of different fragments; (1), (2) and (3) are the pairwise matching results of the gargoyle models; (4), (5) and (6) are the results of the cake models; (7), (8) and (9) are the results of the venus models.

different fragment models including cake, gargoyle and sculpture in the comparison experiments. The results are shown in Figure 10, where different rows correspond to the pairwise matching results of different models. Columns (a) and (b) represent the original model, columns (c) and (d) show the reassembly results obtained by the methods in [18] and [19], respectively. Our pairwise matching results are shown in the column (e).

It can be observed from Figure 10 (c) that the pairwise matching method in [18] produces obvious errors. For example, there are obvious gaps in the matching results of the cake model and the first two gargoyle model. The results for the sculpture model also have obvious deviations. The reason is because the method in [18] uses the contour lines as the features for matching. If the contour is not obvious or missing, the matching accuracy will be affected. For the same fragment model in Figure 10 (d), the method in [19] has a penetration phenomenon, which makes it difficult to achieve better matching results. The method in [19] only uses the thickness feature of fragments for matching, and requires the spatial density of the two aligned sequences in the LCS algorithm must be the same, which is the reason why the fragments cannot be matched correctly. It can be clearly seen from Figure 10 that the cake model is partially missing, the gargoyle model is partially matched, and the sculpture model has rich surface features. In general, no matter whether the feature of fragment is obvious or not, our method can achieve better matching results compared with other methods.

D. RESULTS OF ELIMINATING MISMATCHES

This article uses the improved RANSAC to eliminate mismatches. In order to evaluate the performance of the proposed method in mismatch elimination, we conducted the experiments to compare the improved RANSAC with the conventional geometric consistency algorithm [35]. The results are shown in Figure 11, where (a), (b), (c) are the results obtained by geometric consistency while (d), (e) and (f) are the results obtained by the improved RANSAC.

As can be seen from Figure 11, our improved RANSAC eliminates the pairs of mismatched points while there are still mismatching pairs (yellow lines) after running the geometric consistency algorithm, which misses mismatching pairs due to its limitations of only using the Euclidean distance and the angle constraint between two groups of matching pairs as

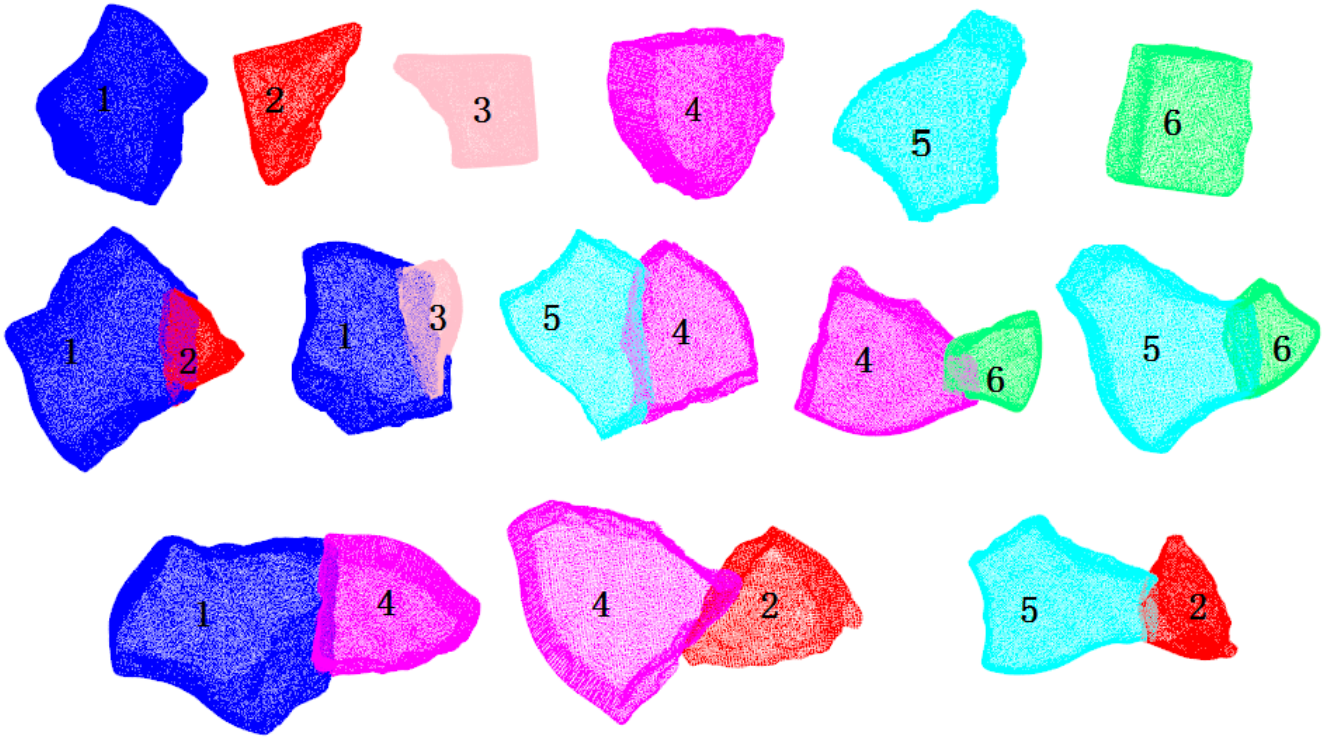


FIGURE 9. Fragments of brick and their pairwise matching.

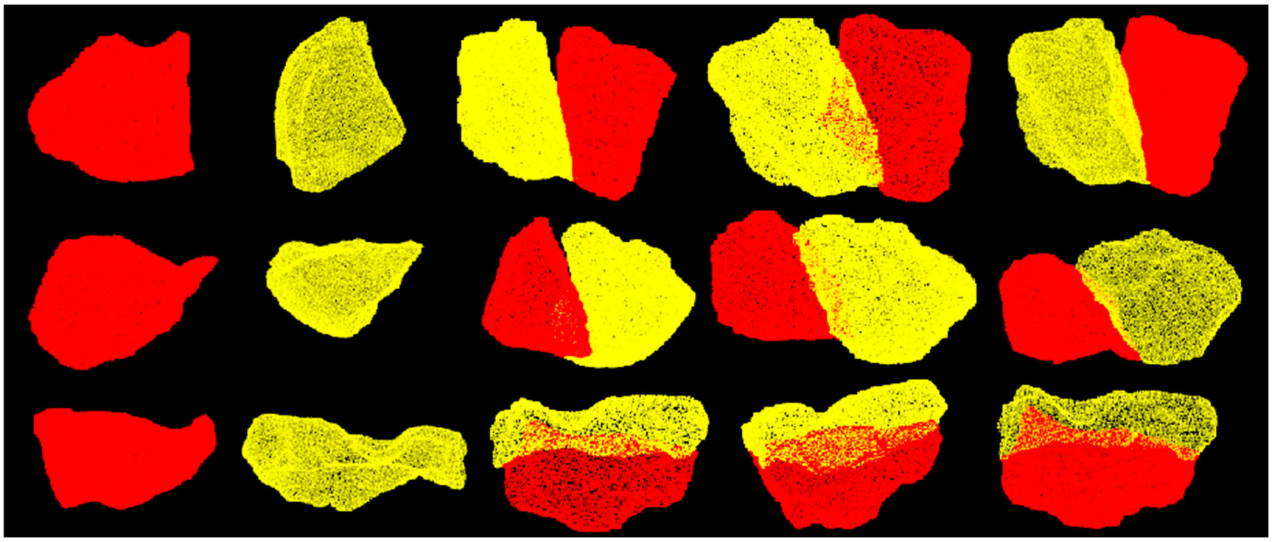


FIGURE 10. Comparing our pairwise matching method with the existing methods. (a), (b) Original models; (c) Matching results from the method in [18]; (d) Matching results from the method in [19]; (e) Matching result of our method.

the eliminating conditions. The results from Figure 11 show that the method proposed in this article can eliminate the mismatching pair effectively, which further improves the correctness of 3D fragment reassembly.

TABLE 3 lists the detailed experimental records of eliminating the mismatches. The assessment criteria include the number of correct matching pairs, the number of mismatching pairs, and the correct matching ratio obtained by the methods for the two groups of point clouds. As can been seen from TABLE 3, compared with geometric consistency, our improved RANSAC has improved mismatch detection significantly.

$$correct\ matching\ rate = \frac{correct\ matching\ pairs}{correct\ matching\ pairs + mismatching\ pairs}$$

TABLE 4 compares our improved RANSAC with more existing methods for eliminating mismatches, including those

TABLE 3. Experimental comparison of mismatching elimination.

Assessment index	Geometric Consistency			Improved RANSAC		
	a	b	c	d	e	f
Correct matching pairs	12	35	43	18	19	26
Mismatching pairs	2	11	21	0	1	2
Correct matching ratio	0.857	0.760	0.67	1	0.95	0.928

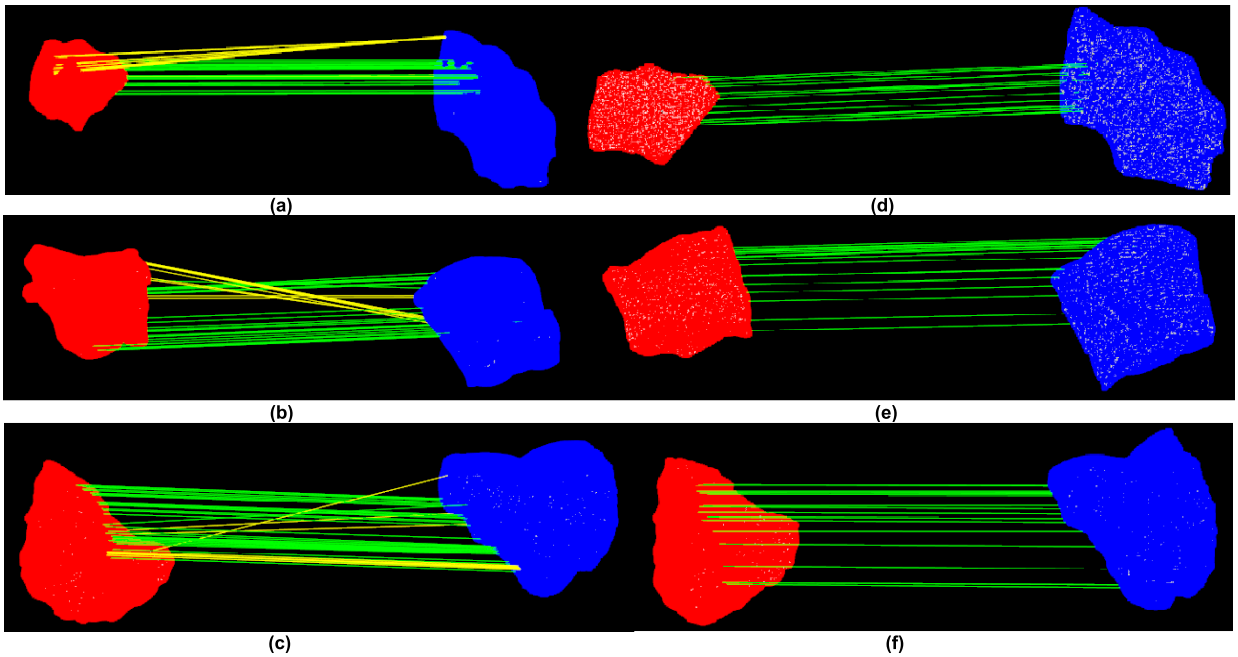


FIGURE 11. Compare the improved RANSAC with the geometric consistency algorithm in terms of mismatch elimination; (a), (b), (c) show the results of eliminating the mismatching using geometric consistency while (d), (e), (f) are the results using our improved RANSAC; the green and yellow lines represent the correct wrong matching, respectively; (a) and (d) are for the cake models, (b) and (e) are for the sculpture models, (c) and (f) are for the gargoyle models.

in [24], [26] and [27]. We use Root Mean Square Error (RMSE, representing the accuracy) and Execution Time as the assessment criteria. Different rows in the table are for the corresponding fragment models in Figure 8. The nine *Test* models in TABLE 4 correspond to the nine models in Figure 8.

Our improved RANSAC invokes Algorithm 1 to perform sampling and evaluate the initial estimation for the transformation and then calls Algorithm 2 to eliminate mismatches. We design Algorithm 1 to help our method to achieve excellent matching accuracy while designing Algorithm 2 to eliminate the mismatches efficiently. In order to evaluate the effectiveness of Algorithm 1 and Algorithm 2 separately, we also ran our method with only Algorithm 1 switched on while the mismatch elimination is performed by the operations in the conventional RANSAC. This method is labelled as ''Our-1'' in TABLE 4. Our improved RANSAC, i.e., the method with both Algorithm 1 and Algorithm 2 switched on, are labelled as ''Our-2'' in TABLE 4.

It can be seen from TABLE 4 that our improved RANSAC (i.e., Our-2) is much faster than these existing methods under the same convergence condition. As reflected by the values of RMSE, the accuracy of our algorithm is also higher than that of other three algorithms, thanks to Algorithm 1.

When comparing ''Our-1'' and ''Our-2'' in TABLE 4, we can see that ''Our-2'' achieves the similar accuracy as ''Our-1'', and takes much less time than ''Our-1''. These results verify that Algorithm 1 can improve the matching accuracy while Algorithm 2 can eliminate the mismatches efficiently, which significantly reduces the execution time of the method.

E. RESULTS OF GLOBAL REASSEMBLY

Our method can perform not only the pairwise matching, but also the global reassembly of multiple fragments. A global reassembly is obtained by repeating the pairwise matching process until all fragments are matched. When a match is aligned, the pairwise fragments are regarded as a new

TABLE 4. Comparing the pairwise matching results obrained by our algorithm and the existing algorithms in literature.

Test	RMSE(mm)					Execution Time(sec)				
	[24]	[26]	[27]	our-1	our-2	[24]	[26]	[27]	our-1	our-2
(1)	0.072	0.053	0.027	0.021	0.023	8.52	7.34	6.31	12.10	3.62
(2)	0.127	0.103	0.117	0.103	0.103	9.22	9.05	6.27	13.26	3.78
(3)	0.352	0.243	0.271	0.236	0.236	8.34	8.15	7.34	14.62	4.35
(4)	0.397	0.314	0.306	0.260	0.260	10.13	9.57	8.06	16.84	3.27
(5)	0.452	0.391	0.324	0.319	0.319	10.34	9.61	7.52	15.16	4.81
(6)	0.569	0.476	0.453	0.420	0.420	7.52	7.05	6.91	13.12	2.83
(7)	0.612	0.586	0.507	0.437	0.437	10.22	9.68	7.58	16.74	2.03
(8)	0.446	0.403	0.412	0.402	0.402	12.93	11.25	8.23	19.31	2.17
(9)	0.357	0.301	0.255	0.247	0.247	8.47	7.83	6.34	15.28	2.56

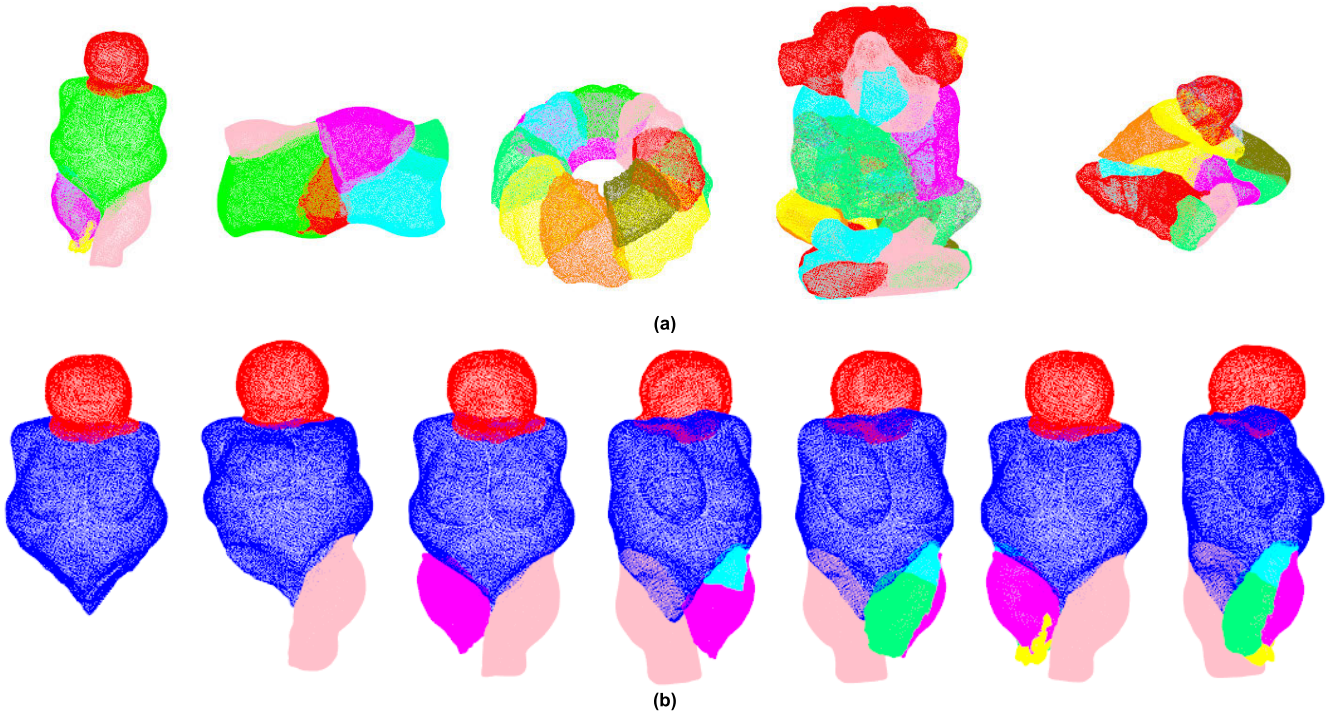


FIGURE 12. The global assembly of models; (a) The reassembling results for the models of venus, brick, cake, gargoyle and sculpture. (b) The global reassembly of incomplete venus fragments.

fragment, which is added to the candidates. Then the pairwise matching and aligning continue with the remaining candidates. The results of global reassembly of multiple models are presented in Figure 12(a). It can be seen from the figure that the fragments of all models are reassembled effectively. In practical applications, it is very common that the fragments are incomplete or missing due to severely corrosion. Figure 12(b) shows the results of the global reassembly of the fragments of venus in the cases where some fragments are missing. As can be observed from this figure, our method

can still find all potential fragment pairs in the remaining fragments.

VI. CONCLUSION

A fast fragment reassembling method based on covariance descriptors is proposed in this article. In particular, the keypoints extraction algorithm based on the multi-scale symbolic projection distance is proposed. The concavity and convexity of the keypoints are described. Next, a multi-scale Covariance Descriptor is proposed for better description of the features of keypoint neighborhoods, and the initial matching is then performed according to the similarity of the distance of the covariance descriptor. Finally, the improved RANSAC algorithm is proposed to eliminate the mismatches and achieve the perfect matching of the fragments. The effectiveness of the proposed method is evaluated through extensive experiments. The results show that comparing with the existing methods in literature, our method can restore the original form of the fragments and achieve precise spatial alignment of the fragments more effectively and more efficiently.

REFERENCES

[1] H. ElNaghy and L. Dorst, "Geometry based faceting of 3D digitized archaeological fragments," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Venice, Italy, Oct. 2017, pp. 2934–2942, doi: 10.1109/ICCVW.2017.346.

[2] W. Yang, Z. Mingquan, Z. Pengfei, and G. Guohua, "Matching method of cultural relic fragments constrained by thickness and contour feature," *IEEE Access*, vol. 8, pp. 25892–25904, 2020, doi: 10.1109/ACCESS.2020.2969995.

[3] Y. Zhang, K. Li, X. Chen, S. Zhang, and G. Geng, "A multi feature fusion method for reassembly of 3D cultural heritage artifacts," *J. Cultural Heritage*, vol. 33, pp. 191–200, Sep. 2018, doi: 10.1016/j.culher.2018.03.001.

[4] N. Yu, Z. H. Li, and Z. Yu, "Survey on encoding schemes for genomic data representation and feature learning—From signal processing to machine learning," *Big Data Mining Anal.*, vol. 1, no. 3, pp. 191–210, Sep. 2018, doi: 10.26599/BDMA.2018.9020018.

[5] L. Vlachopoulos, G. Székely, C. Gerber, and P. Fürnstahl, "A scale-space curvature matching algorithm for the reconstruction of complex proximal humeral fractures," *Med. Image Anal.*, vol. 43, pp. 142–156, Jan. 2018, doi: 10.1016/j.media.2017.10.006.

[6] L. Wang, J. Pan, and Q. Yao, "Virtual reassembly of fractured bones for orthopedic surgery," in *Proc. Int. Conf. Virtual Reality Visualizat. (ICVRV)*, Qingdao, China, Oct. 2018, pp. 20–27, doi: 10.1109/ICVRV.2018.00012.

[7] Y. Yan and S. Huang, "A non-redundant benchmark for symmetric protein docking," *Big Data Mining Anal.*, vol. 2, no. 2, pp. 92–99, Jun. 2019, doi: 10.26599/BDMA.2018.9020035.

[8] O. Fabio, S. Anderson, and F. Marcelo, "Efficient 3D objects recognition using multifoveated point clouds," *Sensors*, vol. 18, no. 7, p. 2032, 2018, doi: 10.3390/s18072302.

[9] G. Oxholm and K. Nishino, "A flexible approach to reassembling thin artifacts of unknown geometry," *J. Cultural Heritage*, vol. 14, no. 1, pp. 51–61, Jan. 2013, doi: 10.1016/j.culher.2012.02.017.

[10] Z. Shunyi, H. Rongyong, W. Zheng, and L. Jian, "Reassembling 3D thin fragments of unknown geometry in cultural heritage," *Photogrammetrie-Fernerkundung-Geoinf.*, vol. 2015, no. 3, pp. 215–230, Jun. 2015, doi: 10.1127/pfg/2015/0267.

[11] N. A. Rasheed and M. J. Nordin, "Classification and reconstruction algorithms for the archaeological fragments," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 32, no. 8, pp. 883–894, Oct. 2020.

[12] E. Sizikova and T. Funkhouser, "Wall painting reconstruction using a genetic algorithm," *J. Comput. Cultural Heritage*, vol. 11, no. 1, pp. 1–17, Jan. 2018, doi: 10.1145/3084547.

[13] D. Tsiafaki, A. Koutsoudis, F. Arnaoutoglou, and N. Michailidou, "Virtual reassembly and completion of a fragmentary drinking vessel," *Virtual Archaeol. Rev.*, vol. 7, no. 15, pp. 67–76, 2015.

[14] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann, "Reassembling fractured objects by geometric matching," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 569–578, Jul. 2006, doi: 10.1145/1141911.1141925.

[15] J. Pokrass, A. Bronstein, and M. Bronstein, "Partial shape matching without point-wise correspondence," *Numer. Math.: Theory, Methods Appl.*, vol. 6, no. 1, pp. 223–244, 2013, doi: 10.1017/S1004897900001203.

[16] E. Altantsetseg, K. Matsuyama, and K. Konno, "Pairwise matching of 3D fragments using fast Fourier transform," *Vis. Comput.*, vol. 30, nos. 6–8, pp. 929–938, Jun. 2014, doi: 10.1007/s00371-014-0959-9.

[17] C. Sanchez-Belenguer and E. Vendrell-Vidal, "An efficient technique to recompose archaeological artifacts from fragments," in *Proc. Int. Conf. Virtual Syst. Multimedia (VSMM)*, Hong Kong, Dec. 2014, pp. 337–344, doi: 10.1109/VSMM.2014.7136671.

[18] W. Dongjuan, "The research on pieces of bronze relics virtual stitching technology and geometric feature extraction," M.S. thesis Dept. Comput. Sci., North Univ. Taiyuan, Taiyuan, China, 2017.

[19] L. Chaohua, *Fragment Matching of Terracotta Warriors Based on Their Thickness Feature*. Kirkland, WA, USA: Northwest Univ., 2018.

[20] T.-G. Son, J. Lee, J. Lim, and K. Lee, "Reassembly of fractured objects using surface signature," *Vis. Comput.*, vol. 34, no. 10, pp. 1371–1381, Oct. 2018, doi: 10.1007/s00371-017-1419-0.

[21] M. Wu and J. Wang, "Reassembling fractured sand particles using fracture-region matching algorithm," *Powder Technol.*, vol. 338, pp. 55–66, Oct. 2018, doi: 10.1016/j.powtec.2018.06.045.

[22] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981, doi: 10.1016/b978-0-08-051581-6.50070-2.

[23] F. Hui, K. N. Mu, X. M. Zhao, and J. Y. Ma, "An improved RANSAC algorithm based on the geometric constraints," in *Proc. Int. Conf. Comput. Methods Sci. Eng. (ICCMSE)*, 2014, p. 1618, doi: 10.1063/1.4897763.

[24] Z. Hossein-Nejad and M. Nasri, "An adaptive image registration method based on SIFT features and RANSAC transform," *Comput. Electr. Eng.*, vol. 62, pp. 524–537, Aug. 2017, doi: 10.1016/j.compeleceng.2016.11.034.

[25] S. Choi, T. Kim, and W. Yu, "Performance evaluation of RANSAC family," in *Proc. Brit. Mach. Vis. Conf.*, London, U.K., 2009, pp. 1–12, doi: 10.5244/C.23.81.

[26] J. Lu, H. Shao, W. Wang, Z. Fan, and G. Xia, "Point cloud registration method based on keypoint extraction with small overlap," *Trans. Beijing Inst. Technol.*, vol. 40, no. 4, pp. 409–415, 2020.

[27] K. Zhao, Y. Zhu, and F. Xie, "Point cloud keypoint matching based on improved RANSAC," *Intell. Comput. Appl.*, vol. 8, no. 6, pp. 18–21, 2018.

[28] S. Y. Chu and Z. H. Xi, "Digital image stabilization based on SURF," *J. Comput.-Aided Des. Comput. Graph.*, vol. 26, no. 2, pp. 241–247, 2014.

[29] Y. J. Jia, F. G. Xiong, and X. Han, "Multi-scale keypoint detection based on SHOT," *Laser Optoelectron. Prog.*, vol. 55, no. 7, 2018, Art. no. 071013.

[30] Y.-Y. Zhang, Z.-P. Wang, and X.-D. Lv, "Saliency detection via sparse reconstruction errors of covariance descriptors on Riemannian manifolds," *Circuits, Syst., Signal Process.*, vol. 35, no. 12, pp. 4372–4389, Dec. 2016, doi: 10.1007/s00034-016-0267-x.

[31] F. Xiong and X. Han, "A 3D surface matching method using keypoint-based covariance matrix descriptors," in *IEEE Access*, vol. 5, pp. 14204–14220, 2017, doi: 10.1109/ACCESS.2017.2727066.

[32] B. He, Z. Lin, and Y. F. Li, "An automatic registration algorithm for the scattered point clouds based on the curvature feature," *Opt. Laser Technol.*, vol. 46, pp. 53–60, Mar. 2013, doi: 10.1016/j.optlastec.2012.04.027.

[33] W. Förstner and M. Boudewijn, "A metric for covariance matrices," in *Geodesy-The Challenge of the 3rd Millennium*. Berlin, Germany: Springer, 2003, pp. 299–309, doi: 10.1007/978-3-662-05296-9_31.

[34] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magn. Reson. Med.*, vol. 56, no. 2, pp. 411–421, Aug. 2006, doi: 10.1002/mrm.20965.

[35] B. Yang, Z. Dong, F. Liang, and Y. Liu, "Automatic registration of large-scale urban scene point clouds based on semantic feature points," *ISPRS J. Photogramm. Remote Sens.*, vol. 113, pp. 43–58, Mar. 2016, doi: 10.1016/j.isprsjprs.2015.12.005.

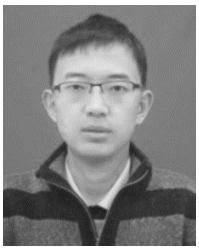
[36] P. Cirujeda, Y. Dicente Cid, X. Mateo, and X. Binefa, "A 3D scene registration method via covariance descriptors and an evolutionary stable strategy game theory solver," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 306–329, Dec. 2015, doi: 10.1007/s11263-015-0820-2.

[37] J. Han, F. Wang, Y. Guo, C. Zhang, and Y. He, "An improved RANSAC registration algorithm based on region covariance descriptor," in *Proc. Chin. Autom. Congr. (CAC)*, Wuhan, China, Nov. 2015, pp. 746–751, doi: 10.1109/CAC.2015.7382597.
[38] *3D Puzzles-Reassembling Fractured Objects by Geometric Matching*. Accessed: Aug. 20, 2018. [Online]. Available: http://www.geometrie.tuwien.ac.at/ig/3dpuzzles.html

CAIQIN JIA is currently pursuing the Ph.D. degree in system simulation and modeling with the School of Data Science and Technology, North University of China, China. Her current research interests include computer graphics, 3-D object reassembly, and virtual reality.

LIGANG HE (Member, IEEE) is currently a Reader with the Department of Computer, University of Warwick. He has authored or coauthored more than 130 articles in international conferences and journals, such as the IEEE TC, TPDS, TACO, IPDPS, SC, and VLDB. His research interests focus on parallel and distributed processing and big data processing.

XIAOWEN YANG received the Ph.D. degree in communication and information system from the North University of China, China, in 2020. She is also an Associate Professor with the School of Data Science and Technology, North University of China. Her research interests focus on computer graphics, 3-D model editing, and virtual reality.



XINGCHENG HAN received the Ph.D. degree from the North University of China, China, in 2018. He is currently a Lecturer with the School of Information and Communication Engineering, North University of China. His research interests include signal processing and recognition, automatic detection technology, and information reconstruction.

BOBO CHANG is currently pursuing the M.S. degree in system simulation and modeling with the School of Data Science and Technology, North University of China, China. His current research interests include computer graphics, 3-D fragment reassembly, and virtual reality.

XIE HAN received the master's degree in computer science and technology from the North China Institute of Technology, China, and the Ph.D. degree from the Institute of Information Engineering, University of Science and Technology Beijing, China, in 2002. She is currently a Professor with the School of Data Science and Technology, North University of China. Her current research interests include computer vision, simulation, and visualization.