Dictionary optimization and constraint neighbor embedding-based dictionary mapping for superdimension reconstruction of porous media

Yang Li, Qizhi Teng,* Xiaohai He, Chao Ren, Honggang Chen, and Junxi Feng
College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China

(Received 17 March 2019; published 27 June 2019)

The three-dimensional (3D) structure of a digital core can be reconstructed from a single two-dimensional (2D) image via mathematical modeling. In classical mathematical modeling algorithms, such as multipoint geostatistics algorithms, the optimization of pattern sets (dictionaries) and the mapping problems are important issues. However, they have rarely been discussed thus far. Pattern set (dictionary)-related problems include the pattern set (dictionary) size problem and the one-to-many mapping problem in a pattern set (dictionary). The former directly affects the completeness of the dictionary, while the latter is manifested such that a single to-be-matched 2D patch has multiple matching patterns in the library and it is hence necessary to select these modes to establish an optimal mapping relationship. Whether the two above-mentioned problems can be properly resolved is directly related to the accuracy of the reconstruction results. Super-dimension reconstruction is a new 3D reconstruction method proposed by introducing the concepts of training dictionary, prior model, and mapping into the reconstruction of the digital core from the field of super-resolution reconstruction. In addition, mapping relationship extraction and dictionary building are also key issues in super-dimension reconstruction. Therefore, this paper discusses these common dictionary-related problems from the perspective of super-dimension dictionaries. We propose dictionary optimization using augmentation dictionaries and clustering based on the boundary features of the dictionary elements to improve the completeness and expand the expression ability of the dictionary. Furthermore, we propose constraint neighbor embedding-based dictionary mapping to establish a more reasonable dictionary mapping relationship for super-dimension reconstruction, and we solve the one-to-many mapping problem in the dictionary. Our experimental results show that the performance of the super-dimension dictionary can be improved by the above-mentioned algorithm. Thus, through the optimized dictionary structure and mapping relationship determined by the above-mentioned methods, the 2D patch to be reconstructed can match a more accurate 3D block in the dictionary. Consequently, the reconstruction precision is improved.

DOI: 10.1103/PhysRevE.99.062134

I. INTRODUCTION

Numerical simulation is an effective method to obtain a three-dimensional (3D) model of the core. In the simulation, the 3D model of the core can often be reconstructed from the two-dimensional (2D) slice image based on different statistical information and geometric features. According to the research in recent years, numerical simulation algorithms are divided into stochastic reconstruction methods, multi-point geostatistical algorithms and machine learning approaches.

Stochastic reconstruction methods are a common method. Initially, Adler *et al.* [1] introduced stochastic reconstructions to the reconstruction field which is based on Gaussian random field technique. After that, there is a correlation function based stochastic reconstruction methods. In the application, two-point correlation function and linear path function, etc. are generally used as the objective function in the reconstruction process. Through continuous iterative exchange, the objective function of the reconstructed image gradually converges to the objective function of the training image. In 1997, Yeong *et al.* [2] proposed to use two-point probability function and lineal-path function obtained from 2D cuts for reconstruction of 3D core. In 2015, Gerke *et al.* [3] proposed a weighting scheme of the objective functions which led to complete accurate reconstructions. In 2018, Karsanina *et al.* [4] proposed a novel hierarchical annealing method based on rescaled correlation functions to improve both accuracy and computational efficiency of reconstructions, which can solve the universal upscaling or downscaling problem.

The multipoint geostatistics algorithm, so called owing to its initial use in the geological simulation process, is nowadays more frequently referred to as the multi-point statistics (MPS) algorithm [5–11]. It is categorized as a typical pattern-matching algorithm. In the simulation process, the pattern library is first established by point-by-point scanning of the 2D training image. Then, the value of the point to be simulated is determined by searching for the best matching pattern from the pattern library on the basis of the known points in the structure to be reconstructed. Blunt *et al.* [10,11] introduced this algorithm into the field of digital core 3D reconstruction. Over the years, many researchers have done a lot of research work on template size problems and reconstruction speed issues. Strebelle *et al.* [6] proposed the SNESIM (single normal equation simulation) algorithm, which uses the structure of the search tree to build a pattern set to speed up the search.

*qzteng@scu.edu.cn

Straubhaar et al. proposed the IMPALA (improved parallel using list approach) algorithm [12], which uses a list to storage pattern set and uses the structure of the search tree to organize the pattern set, which greatly improves the matching speed during reconstruction.

Machine-learning-based approaches are new reconstruction methods that have emerged in the recent years with the prevalence of deep learning. In 2017, Mosser *et al.* [13] used GAN (generative adversarial neural networks) to generate structures similar to 3D core samples. In 2018, Feng *et al.* [14] proposed to combine the deep learning with the traditional three-step sampling algorithm to improve the reconstruction speed.

The pattern set (dictionary)-related problem in the field of 3D reconstruction of porous media has rarely been discussed thus far. Classical reconstruction algorithms, including multipoint geostatistics, involve pattern set (dictionary)-related problems.

Pattern set (dictionary)-related problems include the pattern set (dictionary) size problem and the one-to-many mapping problem in a pattern set (dictionary). The former directly affects the completeness of the dictionary, while the latter is manifested such that a single to-be-matched 2D patch has multiple matching patterns in the library and it is hence necessary to select these modes to establish an optimal mapping relationship. These two problems are related to whether the dictionary matching result in the reconstruction stage is reasonable, and whether the reconstruction result is accurate, which are important issues.

Previously, we proposed super-dimension (SD) reconstruction [15,16], a 3D reconstruction method for porous media. The SD reconstruction theory extends the resolution enhancement principle of the super-resolution (SR) algorithm to dimension enhancement between 2D and 3D porous media, and it introduces concepts such as mapping relationships, prior models, and training dictionaries from learning-based SR. In the training stage, the existing 3D imaging equipment can be used to obtain the 3D image of the core as the source of the 3D prior information. Then, a large amount of sample training is performed to establish the mapping relationships between any 2D patch (such as 3×3) and the corresponding 3D block (such as $3\times3\times3$). In the reconstruction stage, by using the learning mechanism, the input 2D image can reconstruct its 3D structure through this mapping relationship. Compared with resolution improvement, dimension improvement from 2D to 3D is more complicated, and it involves more problems in mapping relationship extraction, dictionary building, block matching, and so on. That is to say, in the theory of SD reconstruction, there are also dictionary-related problems similar to the multi-point geostatistics algorithm.

Our research group has proposed a block matching strategy for the SD algorithm, which improves the reconstruction accuracy [15]. We have also proposed an SD-based 3D nonstationary porous medium reconstruction algorithm [16]. However, problems such as mapping relationship extraction and dictionary establishment remain to be solved. These problems not only exist in SD reconstruction but also are critical problems in multipoint geostatistics algorithms. Therefore, the present paper discusses these common dictionary-related problems from the perspective of SD dictionaries.

From the aspect of building a dictionary, super-resolution [17–20] establishes a high-low resolution mapping dictionary through a large amount of sample training [21–23]. Compared with the correspondence between 2D high- and low-resolution images, the mapping relationship between 2D image patches and 3D image blocks in the SD dictionary is more complicated. Therefore, in the field of SD, it is crucial to explore how to build a reliable dictionary that can satisfy the completeness of the sample and effectively organize the structure of the training sample to establish an optimal mapping.

In this paper, we propose the concept of augmentation dictionaries for the completeness of the SD dictionary, which improves the completeness of the dictionary by self-expanding the existing elements in the dictionary. Then, to further improve the expression ability of the SD dictionary, we propose that the image blocks in the dictionary be divided into horizontal edge blocks, vertical edge blocks, slope edge blocks and non-edge blocks by the clustering algorithm, which facilitates rapid location of the 2D patch to be reconstructed during reconstruction and thus yields the most accurate corresponding 3D block. To establish the optimal mapping relationship for the one-to-many correspondence in the SD dictionary, we introduce the concept of neighborhood embedding in super-resolution reconstruction, and we propose a constraint neighbor embedding algorithm that combines 2D and 3D information according to the characteristics of SD theory.

The remainder of the paper is organized as follows. Section II describes the proposed augmentation dictionaries and clustering based on the boundary features of the dictionary elements. Section III explains the proposed constraint neighbor embedding-based dictionary mapping. Section IV provides a schematic diagram and flowchart of the entire algorithm. Section V presents and analyzes the experimental results. Finally, Sec. VI summarizes the paper.

II. DICTIONARY OPTIMIZATION BY AUGMENTATION DICTIONARIES AND CLUSTERING BASED ON BOUNDARY FEATURES OF DICTIONARY ELEMENTS

A. Augmentation dictionaries

When preparing a dictionary for the SD reconstruction algorithm, different types of CT sequences are selected as the training set. Each CT sequence is traversed in a raster path using an $N \times N$ 2D template in N layers every time. After each traversal, an $N \times N \times N$ 3D block is obtained. The correspondence between the 2D image patch in the first layer and the entire 3D image block is used as an element in the dictionary. The schematic diagram is shown in Fig. 1. Let $\mathrm{D} = \{(d_{2\mathrm{D},1}, d_{3\mathrm{D},1}), \ldots, (d_{2\mathrm{D},i}, d_{3\mathrm{D},i}), (d_{2\mathrm{D},N}, d_{3\mathrm{D},N})\}$ be the constructed dictionary, where $(d_{2\mathrm{D},i}, d_{3\mathrm{D},i})$ denotes the corresponding 2D and 3D samples, i.e., the 2D image patch and its corresponding 3D image block.

As a single 2D image patch usually corresponds to multiple 3D blocks, it is necessary to include as many objectively existing correspondences as possible to ensure the completeness of the dictionary. However, owing to the large number of image samples and the complex relationship between the 2D image patch and its 3D structure, the dictionary cannot usually cover all possible situations. As a result, the 2D patch may match to

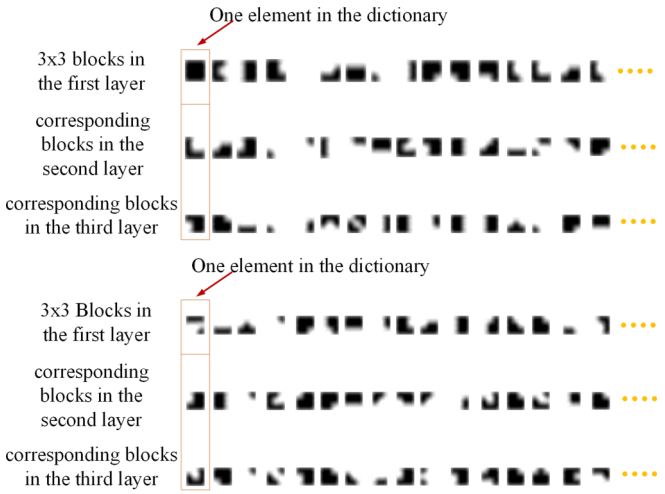

FIG. 1. Dictionary establishment for the SD algorithm.

an unrealistic 3D block in the reconstruction stage, which will affect the accuracy of the reconstructed result. Therefore, how to complete the dictionary is an important research topic.

The completeness of the dictionary can be improved by expanding the training set. However, the obtained 3D training set may be limited, and simply expanding the training set is uneconomical. Therefore, we propose augmentation dictionaries using the elements in the dictionary to improve the completeness of the dictionary. The augmentation dictionaries are constructed as follows. The elements in the existing dictionary are 2D patches and their corresponding 3D blocks. A new element can be generated by rotating a 2D patch and its corresponding 3D block through 0°, 90°, 180°, or 270°. A schematic diagram of the augmentation dictionaries is shown in Fig. 2.

The advantage of this method is that even if the existing training set is small, new dictionary elements can be obtained from the existing dictionary without adding external information. As the existing elements in the dictionary come from real cores, it is conceivable that the rotated elements are also real; more importantly, they are informative and not redundant.

B. Clustering based on boundary features of dictionary elements

The previously established rotation-extended dictionary can improve the completeness of the dictionary, and the dictionary thus becomes more informative. However, an issue that requires further investigation is how to enhance the expressive ability of the dictionary so that a 3D block in line with the real core conditions can be found more accurately in the dictionary for any 2D patch to be reconstructed. Toward this end, clustering the elements in the dictionary according to certain characteristics is an effective method.

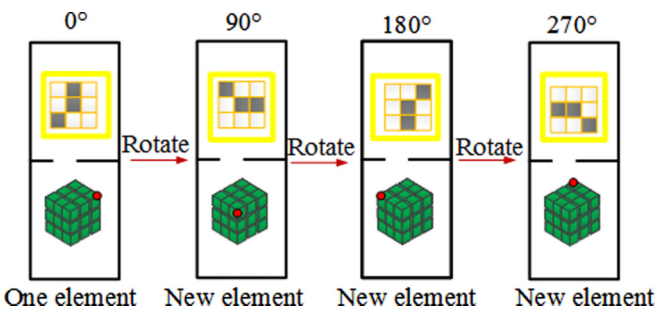

FIG. 2. Schematic diagram of the rotation-extended dictionary.

Clustering is a commonly used data analysis approach in an unsupervised learning environment. Its objective is to divide the data according to the degree of dependence between them to facilitate accurate analysis and extraction of potential laws or patterns hidden in the data. Common cluster analysis techniques include K-means and so on.

We envisage that in the dictionary training of SD reconstruction, clustering analysis is used to cluster the similarity patterns into categories, and the disordered pattern sets are organized by category, which facilitates rapid location of the 2D patch to be reconstructed in the cluster it belongs to during reconstruction. Then, it is more convenient to search for the corresponding 3D block in the cluster. If the number of clusters is set to M, then the entire dictionary is divided into M subspaces according to the cluster center, and the 2D patch and its corresponding 3D pattern are divided into respective clusters as one matching pair of the subspace. The formula is given by

$$\theta^{M}(\text{Patternset}_i) = \psi(\text{Patternset}_i(\text{Pattern } 2d_i)) \quad i = 1, 2, \cdots N \tag{1}$$

where ψ represents the clustering algorithm and M represents the number of categories.

To cluster the elements in the dictionary according to certain characteristics, we need to perform feature selection. The purpose of feature selection is to select features with a large amount of information and to represent the sample itself to a greater extent. The boundary features defined in this paper are such features.

Similar objects have similar boundaries, and the boundary of an object contains important information about the shape of the object. Therefore, in the field of image processing and pattern recognition, an object is often identified or classified according to its boundary. This rule is similarly applicable to the pore boundaries of porous media. After a porous medium is partitioned into patches, the differences among the pixels in some of them are not large; such patches are generally called smooth non-edge patches. For some other image patches that contain the edges of the pores or some other details, the differences among the pixels in the patch are considerable; such patches are called edge patches. A schematic diagram of edge and non-edge patches is shown in Fig. 3. Obviously, the edge patch is rich in high-frequency information. The elements in the dictionary can be well clustered by the boundary features.

As mentioned previously, the elements stored in the dictionary are matching pairs of 2D patches and their corresponding 3D blocks. The basic idea of the proposed clustering algorithm is to classify the elements in the dictionary into different categories by clustering the 2D patches in such elements on the basis of the boundary features.

The first step of the proposed clustering algorithm is to detect the boundary of the 2D patch of the dictionary element to initially determine whether it is a potential patch containing

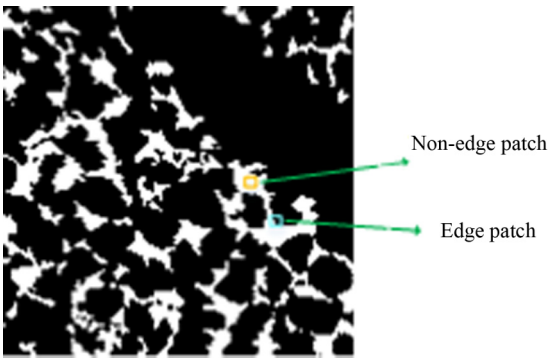

FIG. 3. Schematic diagram of edge and nonedge patches.

a pore boundary. In this step, the elements in the dictionary are divided into elements with non-edge patches and elements with likely edge patches. Here, we use gradient-based edge detection approaches and first-order finite difference approximation to obtain the gradient (rate of change) of the pixel value. Thus, the difference quotient $(\Delta f/\Delta x)$ is approximated to replace the derivative $(\partial f/\partial x)$. In such a method of determining the rate of change of the pixel, the first-order partial derivative of the x and y directions is replaced by the difference between adjacent pixels in the x and y directions. In general, binary images are used in the reconstruction of porous media. For binary images, the method is as follows. Every pixel in the 2D patch (such as 5×5) of one element is traversed. For this pixel, when there are both a pore phase pixel and a rock phase pixel around it, and the pixel itself is a pore phase, it is defined as a potential boundary point. Traversing the entire 2D patch, if there is at least one potential boundary point in the patch, then the element is detected as an element with a likely edge patch; if there is no potential boundary point in the patch, then it is detected as an element with a nonedge patch. A schematic diagram of the detection process is shown in Fig. 4(a). The point indicated by the black box in Fig. 4(a) can be detected as a point on the boundary by the above-mentioned method. The 2D patch (such as 5×5) corresponds to the element defined as an element with a likely edge patch. Fig. 4(b) shows the gradient vector, azimuth, and edge direction of the center point (the edge of any point is orthogonal to the gradient vector).

The second step of the clustering algorithm further subdivides the element with a likely edge patch. The projection length (dimension) of the 2D patch for the element on the horizontal and vertical axes is defined as the threshold Θ (5 for Fig. 4). The total length (number) of the horizontal axis projection points of all the potential boundary points in the 2D patch is defined as $\boldsymbol{\delta}_{\text{row}}$, and the total length (number) of the vertical axis projection points is defined as $\boldsymbol{\delta}_{\text{col}}$. For an element in the dictionary, when $\boldsymbol{\delta}_{\text{row}} = \boldsymbol{\Theta}$ and $\boldsymbol{\delta}_{\text{col}} \neq \boldsymbol{\Theta}$, the element is defined as an element with a horizontal edge patch. When $\boldsymbol{\delta}_{\text{row}} \neq \boldsymbol{\Theta}$ and $\boldsymbol{\delta}_{\text{col}} = \boldsymbol{\Theta}$, the element is defined as an element with a vertical edge patch. When $\boldsymbol{\delta}_{\text{row}} = \boldsymbol{\Theta}$ and $\boldsymbol{\delta}_{\text{col}} = \boldsymbol{\Theta}$, the element is defined as an element with a slope edge patch. When $\boldsymbol{\delta}_{\text{row}} < \boldsymbol{\Theta}$ and $\boldsymbol{\delta}_{\text{col}} < \boldsymbol{\Theta}$, the element is defined as an element with a non-edge patch, as mentioned above. The reason is that if in such a small 2D patch, the total length of the projection of the potential boundary point on the horizontal and vertical axes does not reach the threshold, we consider it to be isolated noise rather than a part of the boundary.

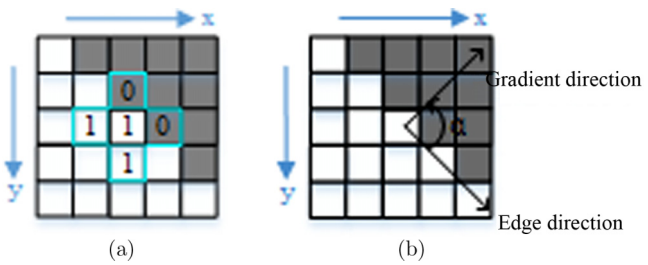

FIG. 4. Edge detection: (a) schematic diagram of the detection process and (b) gradient vector, azimuth, and edge direction of the point to be detected.

During reconstruction, for every 2D patch traversed by the template in the reference image, we first extract the boundary features and perform cluster analysis to classify it into a certain type of the above-mentioned classes. Then, we need not look up the entire dictionary; instead, we can search for the corresponding 3D structure directly in the refined class. This method narrows the range of elements in the dictionary that need to be looked up, thus making the reconstruction result closer to the real situation.

III. CONSTRAINT NEIGHBOR EMBEDDING-BASED DICTIONARY MAPPING

Dictionary performance has been improved by previous methods for improving the dictionary completeness and expressiveness. The rotation-extended dictionary improves the completeness of the dictionary, and cluster analysis of the boundary enables the 2D patch to be reconstructed to accurately locate a certain class of elements in the dictionary. However, another problem to be addressed is the one-to-many mapping relationships in the dictionary, i.e., a single 2D patch may correspond to multiple 3D blocks in the dictionary with a high probability. In this section, we further optimize the dictionary by means of neighborhood embedding to establish a more reasonable mapping relationship.

A. Basic principle of neighborhood embedding in super-resolution (SR)

On the basis of local linear embedded manifold learning, Chang *et al.* [24] proposed a super-resolution (SR) reconstruction method with neighborhood embedding. The method assumes that a high-resolution image block has local geometry similar to that of its corresponding low-resolution image block. According to this assumption, the linear relationship between the input low-resolution image block and the low-resolution image block in the training sample library is obtained and mapped to the high-resolution image block. Thus, the high-resolution image block is reconstructed. Neighborhood embedding-based super-resolution reconstruction can be divided into two main parts: establishment of the sample library and reconstruction.

1. Establishment of the sample library

First, a large number of high-resolution images are selected as training images, and the selected images are degraded to obtain corresponding low-resolution images. There are many degraded models, such as the Gaussian degraded model and the point-average degraded model. The image features are then extracted for the low-resolution digital image using its first- and second-order gradient vectors as the features, and for the high-resolution digital image using the brightness value as the feature. Finally, the high- and low-resolution images are overlapped and segmented, and the high-resolution image block and the corresponding low-resolution image block constitute the corresponding training sample library.

2. Reconstruction

Feature extraction and segmentation are performed on the input low-resolution image. Then, in the low-resolution image block sample library, the most similar k sample blocks are searched for every image block x_t to be reconstructed, and the matching search algorithm uses the Euclidean distance as the nearest neighbor similarity measure function. According to the premise of the method, the high-resolution image blocks corresponding to the searched k low-resolution neighbor blocks can be obtained. Linear combination of k low-resolution neighbor blocks was used to reconstruct the current low-resolution image block, and the optimal reconstruction weight was obtained by minimizing the reconstruction error. The specific formula is given by

$$\omega_t = \arg\min \left\| x_t - \sum_{x_k \in C(x_t)} \omega_k x_k \right\|_2^2 \quad \text{s.t.} \sum_{i=1}^{N} \omega_i = 1 \qquad (2)$$

where ω is the reconstruction weight coefficient, x_k represents the kth neighbor block that is close to the input low-resolution image block x_t, and $C(x_t)$ represents the set of k neighbor blocks of x_t in the training set.

After obtaining the optimal weight, the reconstructed high-resolution image block can be obtained by weighted summation using the weight and the corresponding high-resolution image sample block. The weighted summation formula is as follows:

$$y_t = \sum_{x_k \in C(x_t)} \omega_k y_k, \qquad (3)$$

where y_t is the reconstructed high-resolution image block and y_k is the high-resolution neighbor block corresponding to the low-resolution neighbor block x_k.

In the process of reconstruction from a low-resolution block to a high-resolution block, the method of optimizing the weight is combined with k high-resolution blocks in the training set to minimize the reconstruction error.

B. Drawback of directly using neighborhood embedding in SD reconstruction

The introduction of neighborhood embedding [19,25–30] from super-resolution reconstruction to super-dimension reconstruction has many advantages. (1) For a 2D patch to be reconstructed, if the corresponding 3D block cannot be found in the dictionary such that it completely matches the already reconstructed block boundary, then the approximate optimal solution can be obtained by the neighborhood embedding method. (2) More generally, when the 2D patch to be reconstructed has multiple corresponding 3D blocks in the dictionary, i.e., a one-to-many mapping relationship, we can obtain the optimal mapping relation based on the weighted average of multiple similar neighborhood blocks by using the method of neighborhood embedding. However, directly using neighborhood embedding in super-resolution for super-dimension reconstruction has the following disadvantage.

In super-resolution neighborhood embedding, we have the premise that the weight value of the neighborhood block of the low-resolution block is directly applied to the corresponding high-resolution block, i.e., the high- and low-resolution image blocks have a consistent geometric manifold structure. We can define this aspect as "blindness" in the criterion. However, manifold similarity does not exist between the 2D image patch and the 3D image block in the super-dimension reconstruction process. If super-resolution neighborhood embedding is directly used, then the connection between the 3D block to be reconstructed and the already reconstructed 3D neighborhood block is unnatural. Hence, it is necessary to propose a neighborhood embedding suitable for super-dimension reconstruction.

C. Constraint neighbor embedding-based dictionary mapping for SD reconstruction

To solve the above-mentioned problems, it is necessary to consider how to overcome the blindness of directly applying the optimization weight of the 2D neighborhood patch in the dictionary to the 3D block in super-dimension reconstruction. Hence, we consider not only the information of the 2D patch but also that of the 3D block as the limiting condition when calculating the optimal weight. On the basis of these considerations, we combined the 3D block local constraints and proposed constraint neighbor embedding-based dictionary mapping for SD reconstruction.

The basic idea of the algorithm is as follows. The first step is to determine k 2D-3D matching pairs in the dictionary. For each 2D patch to be reconstructed, after the boundary-based clustering analysis proposed in Sec. II is used to locate a certain type of element in the dictionary, we propose a similarity measure function that combines the 2D and 3D information to determine k 2D-3D matching pairs in the dictionary. The second step is to determine the reconstruction weight. We propose that the distance between the 2D image block to be reconstructed and the 2D sample in the dictionary be taken as one of the constraints, and the distance between the already reconstructed 3D neighborhood block boundary and the 3D sample boundary in the dictionary be taken as another constraint. These two constraints are combined to optimize the reconstruction weight and maintain the image geometry more effectively.

1. Use a similarity measure function to find a 2D-3D matching pair

A new similarity measure function S is proposed. For each 2D patch to be matched, the similarity measure function S is used to search for k 2D-3D matching pairs in the dictionary.

The original super-resolution neighborhood embedding similarity measure function considers only low-resolution block information, which is not adequate when applied to super-dimension reconstruction. When calculating the similarity measure function, we propose that the 2D and 3D information be considered to enhance the correlation of the 2D image patch and the 3D image block and to strengthen the role and weight of the 3D block neighborhood correlation in determining the final 2D-3D matching pair. Specifically, when the 2D patch and the 3D block do not have geometric and manifold similarities, a reasonable 2D-3D matching pair can be found using the neighborhood information and the Markov chain information.

The similarity measure function defined in super-resolution neighborhood embedding is based on the Euclidean distance between the low-resolution blocks. In the super-dimension case, the concept of the similarity measure function defined to find k 2D-3D matching pairs in the dictionary is as follows. Variable l is defined as the Euclidean distance between the 2D patch to be reconstructed and the 2D patch of the matching pair in the dictionary, m is defined as the Markov chain probability of the 3D block of the matching pair in the dictionary, and n is defined as the Euclidean distance between the already reconstructed 3D neighborhood block boundary and the 3D sample boundary of the matching pair in the dictionary. The similarity measure function $S = l \times m \times n$, as shown in Eq. (4). For reconstruction, we calculate the similarity measure function S for any 2D-3D matching pair in the dictionary that has the same type of boundary feature as the 2D patch to be reconstructed. The k 2D-3D matching pairs finally found are the k matching pairs that take S to the maximum value.

$$S = l \times m \times n \tag{4}$$

Its physical meaning is as follows: the k matching pairs that make the variable S reach the maximum value are selected to ensure that the k matching pairs are as similar as possible to the 2D patch to be reconstructed and the already reconstructed 3D neighborhood structure.

The set of k 2D-3D matching pairs found is represented by the symbol D as follows:

$$D = \{(d_{2D,1}, d_{3D,1}), (d_{2D,2}, d_{3D,2}), \ldots, (d_{2D,k}, d_{3D,k})\}. \tag{5}$$

2. Calculate the reconstruction weight

Here, a new reconstruction weight formula for super-dimension neighborhood embedding is proposed. By minimizing the local matching error of the weight formula, the optimization weight ω_i of each 2D patch in the k 2D-3D matching pairs found previously is calculated.

The most basic formula for calculating the reconstruction weight in super-resolution is as follows:

$$\omega_t = \arg\min \left\| x_t - \sum_{x_k \in C(x_t)} \omega_k x_k \right\|_2^2 \tag{6}$$

where ω is the reconstruction weight coefficient, x_k represents the kth neighbor block that is close to the input low-resolution image block x_t, and $C(x_t)$ represents the set of k neighbor blocks of x_t in the training set.

As mentioned previously, when this formula is directly used for super-dimension neighborhood embedding, the geometric structure of the 2D image space is employed and the geometric structure of the 3D image is ignored. As a result, the influence of the 3D block geometry is not well reflected on the weight of reconstruction.

To solve the above-mentioned problem, we propose a new reconstruction weight formula for super-dimension neighborhood embedding as follows:

$$\omega_t = \arg\min \left(\left\| \left(I_{2D} - \sum_{k \in C(I_{2D})} \omega_k d_{2D,k} \right) \right\|_2^2 + \lambda \left\| \left(I_{3D\ \text{boundary}} - \sum_{k \in C(I_{3D\ \text{boundary}})} \omega_k d_{3D\ \text{boundary},k} \right) \right\|_2^2 \right) \tag{7}$$

where ω_k is the reconstruction weight coefficient, $d_{2D,k}$ represents the kth neighboring patch similar to the input 2D image patch I_{2D}, $C(I_{2D})$ represents a set of 2D patches in k 2D-3D matching pairs, $I_{3D\ \text{boundary}}$ represents the already reconstructed 3D block boundary on the input block neighborhood, $d_{3D\ \text{boundary},k}$ represents the boundary of the 3D block corresponding to $d_{2D,k}$ in the dictionary, and $C(I_{3D\ \text{boundary}})$ is the set of 3D image block boundaries corresponding to the 2D image patch in k 2D-3D matching pairs. The distance between $I_{3D\ \text{boundary}}$ and $d_{3D\ \text{boundary},k}$ is taken as another constraint in this equation. Further, λ is a local constrained regularization parameter used to balance the minimization between the reconstruction error and the local prior.

The physical meaning is that the geometric structure information of the already reconstructed 3D blocks is fully utilized, and a weight determination method combining the 3D boundary constraint is proposed. In the weight calculation formula proposed for the super-dimension, the distance between the input 2D patch and the 2D patches in the k 2D-3D matching pairs in the dictionary is taken as one similarity constraint. Further, the distance between the already reconstructed 3D block boundary on the input patch neighborhood and the 3D block boundary in the matching pair is taken as another similarity constraint. The two constraints are combined to better maintain the geometric structure of the block to be reconstructed in 3D space, to further optimize the reconstruction weight, and use to the reconstruction weight to estimate the 3D image block to be reconstructed.

The objective function of the weight formula (here, the number of matching pairs $k = 10$) is defined as follows. For this equation, the derivation process of its elemental expression is given.

$$J(\omega) = \left\| \left(I_{2D} - \sum_{k \in C(I_{2D})} \omega_k d_{2D,k} \right) \right\|_2^2 + \lambda \left\| \left(I_{3D\ \text{boundary}} - \sum_{k \in C(I_{3D\ \text{boundary}})} \omega_k d_{3D\ \text{boundary},k} \right) \right\|_2^2 \tag{8}$$

Assuming that the template used in the dictionary is of size 5 × 5, the matrices I_{2D}, $d_{2D,k}$, $I_{3D\ boundary}$, and $d_{3D\ boundary,k}$ can be expressed as follows:

$$I_{2D} = \begin{pmatrix} I_{2D}(1) & I_{2D}(2) & I_{2D}(3) & I_{2D}(4) & I_{2D}(5) \\ I_{2D}(6) & I_{2D}(7) & I_{2D}(8) & I_{2D}(9) & I_{2D}(10) \\ I_{2D}(11) & I_{2D}(12) & I_{2D}(13) & I_{2D}(14) & I_{2D}(15) \\ I_{2D}(6) & I_{2D}(17) & I_{2D}(18) & I_{2D}(19) & I_{2D}(20) \\ I_{2D}(21) & I_{2D}(22) & I_{2D}(23) & I_{2D}(24) & I_{2D}(25) \end{pmatrix} \tag{9}$$

$$d_{2D,k} = \begin{pmatrix} d_{2D,k}(1) & d_{2D,k}(2) & d_{2D,k}(3) & d_{2D,k}(4) & d_{2D,k}(5) \\ d_{2D,k}(6) & d_{2D,k}(7) & d_{2D,k}(8) & d_{2D,k}(9) & d_{2D,k}(10) \\ d_{2D,k}(11) & d_{2D,k}(12) & d_{2D,k}(13) & d_{2D,k}(14) & d_{2D,k}(15) \\ d_{2D,k}(16) & d_{2D,k}(17) & d_{2D,k}(18) & d_{2D,k}(19) & d_{2D,k}(20) \\ d_{2D,k}(21) & d_{2D,k}(22) & d_{2D,k}(23) & d_{2D,k}(24) & d_{2D,k}(25) \end{pmatrix} \tag{10}$$

$$I_{3D\ boundary} = \begin{pmatrix} I_{3D\ boundary}(1) & I_{3D\ boundary}(2) & I_{3D\ boundary}(3) & I_{3D\ boundary}(4) & I_{3D\ boundary}(5) \\ I_{3D\ boundary}(6) & I_{3D\ boundary}(7) & I_{3D\ boundary}(8) & I_{3D\ boundary}(9) & I_{3D\ boundary}(10) \\ I_{3D\ boundary}(11) & I_{3D\ boundary}(12) & I_{3D\ boundary}(13) & I_{3D\ boundary}(14) & I_{3D\ boundary}(15) \\ I_{3D\ boundary}(16) & I_{3D\ boundary}(17) & I_{3D\ boundary}(18) & I_{3D\ boundary}(19) & I_{3D\ boundary}(20) \\ I_{3D\ boundary}(21) & I_{3D\ boundary}(22) & I_{3D\ boundary}(23) & I_{3D\ boundary}(24) & I_{3D\ boundary}(25) \end{pmatrix} \tag{11}$$

$$d_{3D\ boundary,k} = \begin{pmatrix} d_{3D\ boundary,k}(1) & d_{3D\ boundary,k}(2) & d_{3D\ boundary,k}(3) & d_{3D\ boundary,k}(4) & d_{3D\ boundary,k}(5) \\ d_{3D\ boundary,k}(6) & d_{3D\ boundary,k}(7) & d_{3D\ boundary,k}(8) & d_{3D\ boundary,k}(9) & d_{3D\ boundary,k}(10) \\ d_{3D\ boundary,k}(11) & d_{3D\ boundary,k}(12) & d_{3D\ boundary,k}(13) & d_{3D\ boundary,k}(14) & d_{3D\ boundary,k}(15) \\ d_{3D\ boundary,k}(16) & d_{3D\ boundary,k}(17) & d_{3D\ boundary,k}(18) & d_{3D\ boundary,k}(19) & d_{3D\ boundary,k}(20) \\ d_{3D\ boundary,k}(21) & d_{3D\ boundary,k}(22) & d_{3D\ boundary,k}(23) & d_{3D\ boundary,k}(24) & d_{3D\ boundary,k}(25) \end{pmatrix} \tag{12}$$

Assuming that A is a matrix, $a_{ij}(1 \leqslant I \leqslant m, 1 \leqslant j \leqslant n)$ is an element of matrix A. In the sense of the l_2 norm, the following holds:

$$\|A\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |(a_{ij})|^2 \right)^{\frac{1}{2}}. \tag{13}$$

After derivation, when a 5 × 5 template is used in the dictionary and the number of matching pairs is set to 10, the elemental expression of the objective function is as follows:

$$\mathrm{J}(\omega) = \left(\sum_{i=1}^{25} \left| \left(\mathrm{I}_{2D}(i) - \sum_{k=1}^{10} \omega_k d_{2D,k}(i) \right) \right|^2 \right)^{\frac{1}{2}} + \lambda \left(\sum_{i=1}^{25} \left| \left(\mathrm{I}_{3D\ boundary}(i) - \sum_{k=1}^{10} \omega_k d_{3D\ boundary,k}(i) \right) \right|^2 \right)^{\frac{1}{2}}. \tag{14}$$

The Lagrange multiplier method is used to solve the above-mentioned equation, and the local optimization reconstruction weights $\{\omega_1, \omega_2, \cdots, \omega\}$ can be obtained.

3. Find the optimal mapping in the dictionary through neighborhood embedding

Previously, we obtained k weights $\{\omega_1, \omega_2, \cdots, \omega_k\}$, which are mapped to the k 3D blocks in the 2D-3D matching pairs calculated earlier. Finally, the 3D block I_{3D} corresponding to the input 2D patch is obtained. Its mapping expression is as follows. In the framework of super-dimension reconstruction, a schematic diagram of the dictionary mapping process based on constraint neighborhood embedding is shown in Fig. 5.

$$I_{3D} = \sum_{j=1}^{k} \omega_j d_{3D,j} \tag{15}$$

IV. SCHEMATIC DIAGRAM AND FLOW OF THE OVERALL ALGORITHM

By integrating the second part of the dictionary optimization algorithm and the third part of the dictionary mapping algorithm into the reconstruction process of the original super-dimension, a more complete super-dimension reconstruction algorithm is obtained. A schematic of the reconstruction process and the pseudocode are shown in Fig. 6 and Table I, respectively.

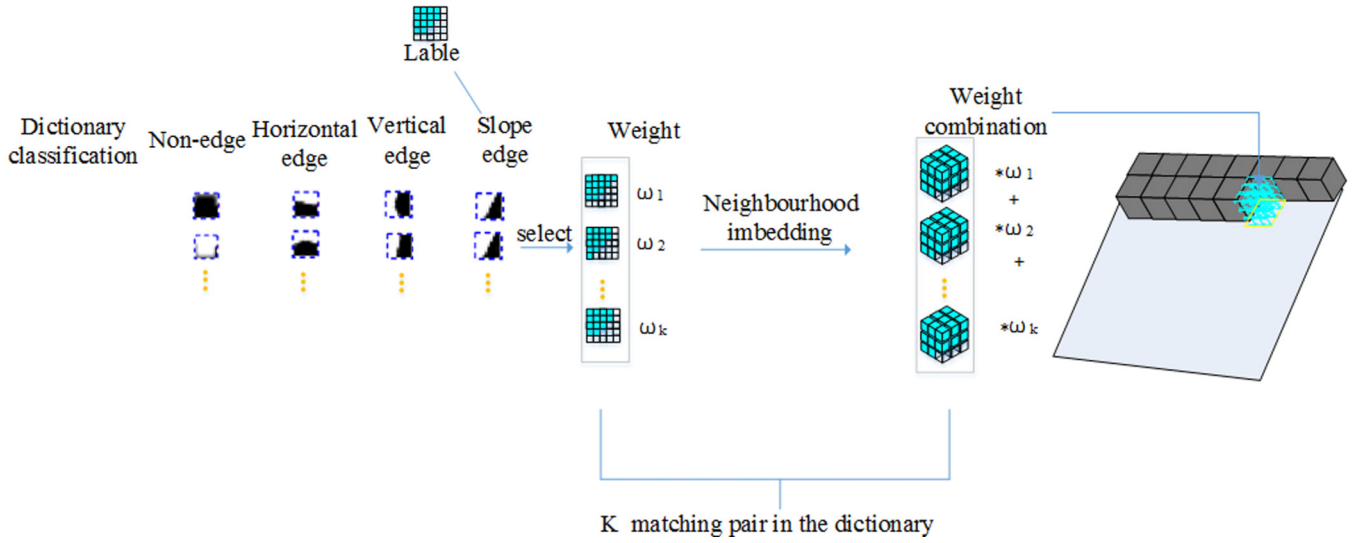

FIG. 5. Schematic diagram of the dictionary mapping process based on constraint neighborhood embedding.

After that, we performed a set 5 repetitive reconstruction experiments for current and conventional SD algorithms, respectively. The reconstruction results for one of the groups are shown in Figs. 7(b), 7(c), and 7(d).

V. RESULTS AND DISCUSSION

To verify the accuracy of the reconstruction results of the SD algorithm with dictionary optimization and dictionary mapping, we performed the following experiment. Figure 7(a) shows a CT image with an image size of 128 × 128 pixels and a porosity of 0.2070, in which the white phase is the pore phase. We performed a set of five repetitive reconstruction experiments to reconstruct its 3D structure with the original and current SD reconstruction algorithms, respectively. The size of the 3D structure reconstructed by the original and current SD algorithms was 128^3 voxels. Simulated with an Intel i7-6700k CPU, the computation time is about 3 h per sample for the original SD algorithm and is about five days per sample for the current SD algorithm. The CT target sample and one of the five sets of experimental results with the original and current SD reconstruction algorithms are shown

TABLE I. Pseudocode of SD reconstruction with dictionary optimization and dictionary mapping.

1. Improve the dictionary completeness by simultaneously rotating the 2D patch and the 3D block of each element matching pair in the dictionary;
2. The dictionary matching pair performs boundary clustering according to the 2D block therein to further expand the expression ability of the dictionary;
3. **For** every 2D block to be reconstructed do
4. Clustering based on boundary features;
5. Clustering the 2D patch into a certain class of elements in the dictionary based on boundary features;
6. // Finding the best matching 3D block in this cluster of the dictionary through the constraint neighbor embedding;
7. **For** the elements in the dictionary
8. Viewing the boundary labels of the 2D block in every element matching pair in the dictionary;
9. If (the boundary type of the 2D patch to be reconstructed = = boundary type of element in dictionary)
10. int S = 0;//Initialize the similarity measure function
11. Calculate and record the similarity measure function corresponding to this matching pair in the dictionary $S = l \times m \times n$; //l, m, n have the same meaning as the variables in Eq. (4)
12. Recording k ($k = 10$) 2D to 3D matching pairs in the dictionary when the similarity measure function S reaches the maximum value;
13. Solve for reconstruction weights;
14. Weighting the k 3D neighborhood blocks found in the dictionary with the solved weights to obtain the final 3D block.

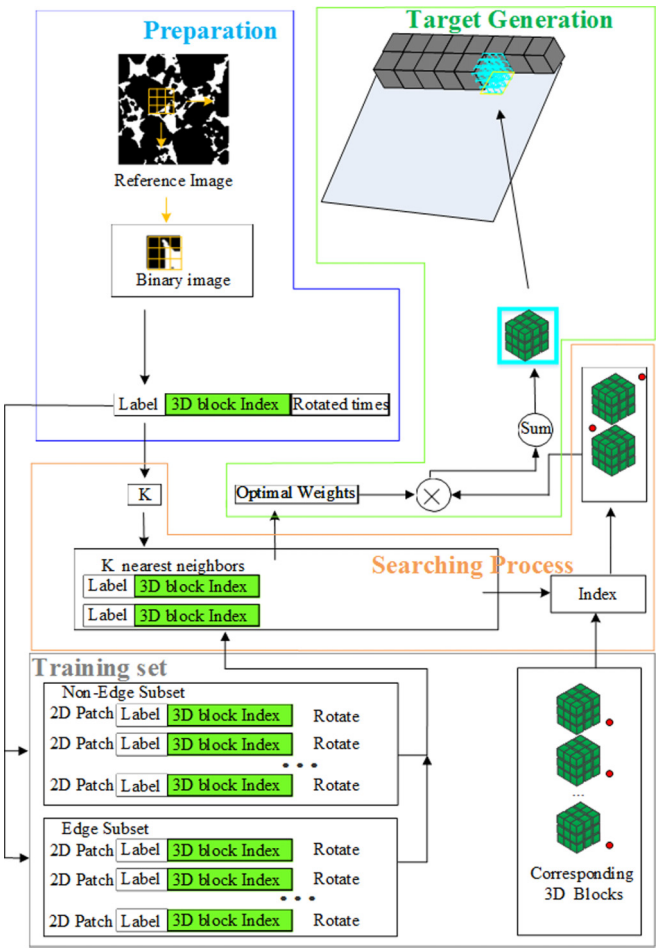

FIG. 6. Schematic diagram of super-dimension reconstruction including dictionary optimization and dictionary mapping algorithm.

in Figs. 7(b), 7(c), and 7(d), respectively. From the visual aspect, the original and current SD algorithms can reproduce the 3D CT target sample well. To further verify that the proposed method improves the accuracy of the reconstruction results compared to the original super-dimension algorithm, we performed a quantitative comparison of different methods.

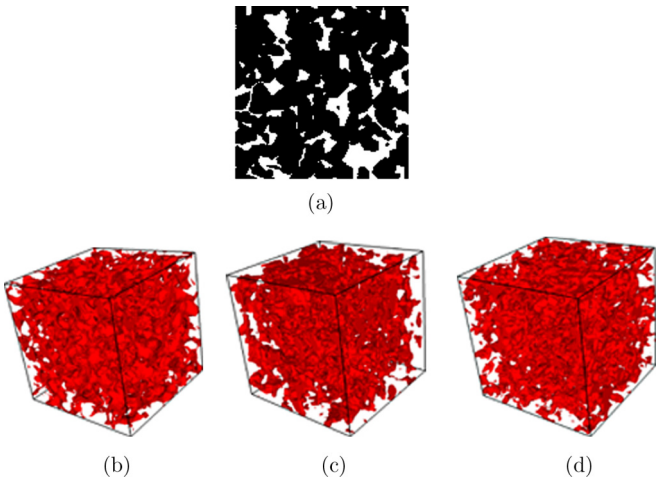

FIG. 7. Three-dimensional reconstructed results of the core: (a) reference image, (b) CT target sample, (c) reconstructed result of the original SD, and (d) reconstructed result of the current SD.

A. Porosity

Porosity is a low-order statistical parameter, but it is important that other high-order parameters of the reconstructed 3D structure match the CT target sample.

First, we compared the porosity of the abovementioned core structures. The porosities of the CT target sample is 0.2047. For the five sets of repetitive experiment, the average porosities of the reconstructed result of the original SD algorithm, and the reconstructed result of the current SD algorithm are 0.2562 and 0.2029, respectively. By analyzing the experimental results, we can conclude that the deviation of the porosity between the reconstructed result of the original SD algorithm and that of the CT target sample is more severe owing to the incompleteness of the dictionary and the uncertainty of the dictionary mapping relationship. However, by using the proposed dictionary optimization and constraint neighbor embedding-based dictionary mapping, the porosity can be controlled effectively.

B. Autocorrelation function

In geostatistics, statistical feature functions, such as the autocorrelation function (ACF) [31–33], are often used to describe the pore space, and the effectiveness of the algorithm is verified by comparing the statistical functions of the reconstructed results and those of the CT target sample.

As the research object of the digital core is generally divided into two parts, the pore and the grain, the phase function $Z(\bar{x})$ of the core can be expressed as follows:

$$Z(\bar{x}) = \begin{cases} 1 & \bar{x} \in \text{pore} \\ 0 & \bar{x} \notin \text{pore} \end{cases} \tag{16}$$

where $\bar{x}$ represents any pixel in the image.

The ACF $S(r)$ represents the probability that two points in a core image in a certain direction separated by distance r are in the pore phase. The mathematical expression is given by

$$S(r) = \overline{Z(x) \times Z(x+r)} \tag{17}$$

where "–" indicates the statistical average. The ACF describes the correlation between pixel points with a distance of r in the image. When $r = 0$, $S(r) = \phi$, where ϕ is the porosity of the image. As r is gradually increased, the correlation between the two points gradually decreases, and when a certain threshold is exceeded, $S(r) \approx \phi^2$. As a perfectly homogeneous core does not exist, $S(r)$ generally performs damped oscillation around ϕ^2. The r value for which $S(r)$ is lower than ϕ^2 for the first time is defined as the autocorrelation distance R.

The normalized ACF curves of the CT target sample and the reconstructed results of the current and original SD algorithms are compared in Fig. 8. From Fig. 8, we can see that the ACF curves of the CT target sample and the reconstructed result of the current SD algorithm are basically consistent. Hence, it can be concluded that the reconstruction result successfully reproduces the statistical information in the

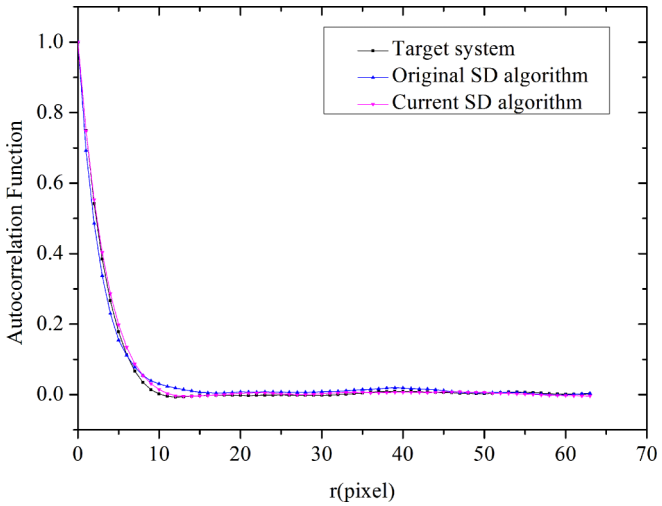

FIG. 8. Comparison of the ACF curves of the CT target sample and reconstructed results of the original and current SD algorithms.

CT target sample. At the same time, it can be seen from the figure that the ACF curve of the current result is closer to that of the CT target sample compared to the original SD algorithm, implying that the structure reconstructed by the current algorithm reflects the main distribution characteristics of the CT target sample more effectively.

C. Euler number

The Euler number can effectively describe the spatial connectivity of 3D structures. It can be expressed as follows:

$$\chi = N_p - N_c + N_h \tag{18}$$

where N_p represents the number of pores in a 3D structure, N_c represents the number of redundant connections inside the pores (number of connected channels), and N_h represents the number of fully closed cavities (where the pores contain grains). If $N_p > N_c$, i.e., the Euler number is positive, then the connectivity of the 3D structure is poor; if $N_p < N_c$, i.e., the Euler number is negative, then the connectivity of the 3D structure is good. Owing to the existence of gravity, in the actual 3D structure of the core, there is no case in which the grains are completely surrounded by the pores; hence,N_h is usually zero.

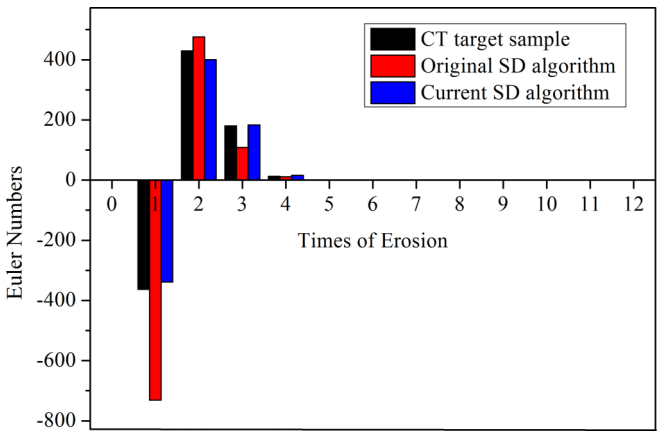

FIG. 9. Comparison of the Euler numbers of the CT target sample and the reconstructed results of the original and current SD algorithms.

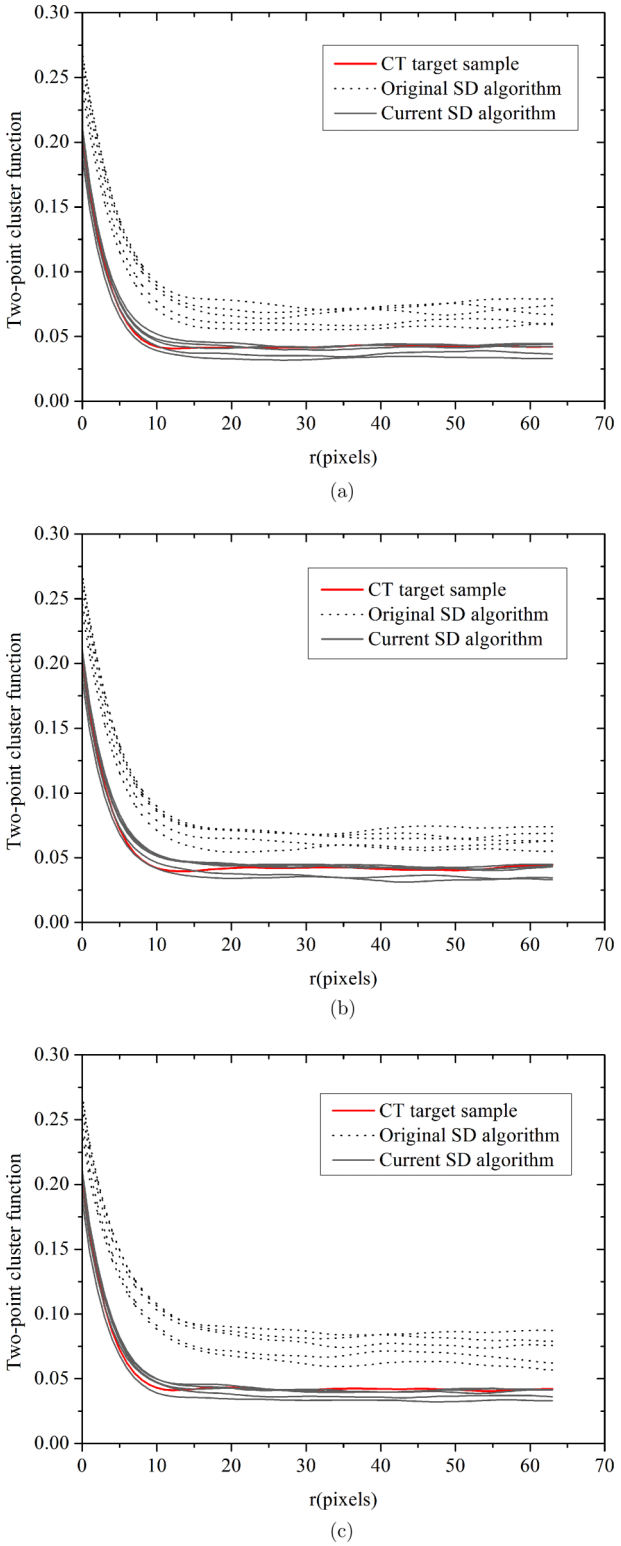

FIG. 10. Comparison of the (a) x-, (b) y-, and (c) z-direction two-point cluster function curves of CT target sample (red line) and the several realizations of the original (dotted line) and current SD (gray line) algorithms.

TABLE II. Comparison of pore morphological parameters.

	CT target sample	Original SD algorithm	Current SD algorithm
Average shape factor	0.0669773	0.0584174	0.0622916
Average size of pore radius (m)	5.53E-05	7.02E-05	6.26E-05
Average size of throat radius (m)	2.08E-05	2.44E-05	2.41E-05
Average volume of pore (m^3)	8.93E-13	1.82E-12	1.26E-12
Average volume of throat (m^3)	1.43E-13	2.85E-13	1.91E-13
Average radius size ratio of pore and throat	0.298306	0.29076	0.292701
Average coordination number	7.67136	9.09774	5.09714

When there are more pore branches in the pore structure, the greater the number of interconnects between the pores, the smaller is the Euler number. By contrast, when there are less pore branches, the greater the number of isolated pores, the larger is the Euler number. In practical applications, Euler numbers are often combined with morphological corrosion operations in image processing. When calculating the Euler number, after performing a corrosion operation on the 3D structure, the Euler number is calculated again. In general, owing to the large number of pore branches in the 3D structure of the core before the corrosion operation, the connectivity is good and the Euler number is small. After the corrosion operation, the finer pore branches in the 3D structure of the core are eliminated, the connectivity is degraded, and the Euler number increases.

Here, the code from Vogel *et al.* was adopted for Euler number computations [34]. The Euler numbers of the CT target sample, the original super-dimension reconstruction results, and the current super-dimension reconstruction results are compared in Fig. 9. It can be seen from the figure that when the number of iterations is 1, the Euler number of the original super-dimension reconstruction result is significantly different from that of the CT target sample. This indicates that the original super-dimension algorithm reconstructs the structure with excessive connections inside the pores. This phenomenon is due to the incompleteness of the dictionary and the one-to-many mapping in the dictionary, which causes errors in the reconstruction. After dictionary optimization and establishment of the optimal mapping, the Euler values of the current super-dimension reconstruction results are closer to those of the CT target sample. Thus, the pore connectivity properties of these two structures are very similar.

D. Two-point cluster function

In a binary image, individual pores are called pore clusters. The two-point cluster function $C_k(\vec{r})$ is defined as follows. Randomly drop a vector of length $\vec{r}$ into a 2D structure or 3D structure, starting from k and ending at l. The two-point cluster function indicates the probability that the starting point and ending point of the vector fall into the same cluster [35,36].

The two-point cluster function uses probability to describe the connectivity between two points in space by counting the frequencies of the two points in the connected cluster that are separated by distance r. The greater the distance r between the two points, and the greater the probability corresponding to it, the better is the connectivity of the pores, and vice versa.

Direction correlation functions can describe the connectivity characteristics of core structures in different directions [37,38]. To further evaluate the algorithm of this paper, Fig. 10 compares the two-point cluster function of the CT target sample with those of the several realizations by the original and current SD algorithms in the x, y, and z directions. From the results of the two-point cluster function curve reflected in the three directions, the current super-dimension reconstruction results have greater consistency with the CT target sample. Thus, the proposed algorithm can better control the connectivity of the pores in the 3D structure and make it consistent with the characteristics of the CT target sample.

E. Pore morphology parameters

On the basis of 3D structural processing techniques, such as the sphere expansion method, 3D geometric transformation technology, and discriminant analysis method, the pore structure is segmented and the area occupied by each pore and throat is marked. Here, the code from Dong and Blunt [39] was adopted to perform pore structure extraction. Through these methods, the pore morphology parameters can be measured, including pore diameter, pore volume, pore shape factor, throat diameter, and throat volume. These parameters reflect the morphological characteristics of the measured 3D structure. The pore morphology parameters of the CT target sample and the reconstruction results of the original and current SD algorithm were calculated, as shown in Table II. It can be seen from the table that the reconstruction result of the proposed algorithm is closer to the 3D structure of the CT target sample in terms of the pore morphology parameters, indicating that the 3D morphology of the reconstruction result is more similar to the CT target sample.

F. Comprehensive analysis of experimental results

The experimental results show that the porosity of the original super-dimension algorithm is greater than that of the CT target sample. The autocorrelation function, Euler number, two-point cluster function, and pore morphology parameters deviate significantly from the CT target sample. There are two reasons for this phenomenon. (1) Error accumulation, i.e., the superposition of some random factors in the reconstruction process, causes the reconstruction structure to always be biased to some fixed mode, resulting in a large deviation between the final reconstructed structure and the CT target sample. (2) A matching error exists because every template corresponds to a large number of modes. For each 2D patch

to be reconstructed, the corresponding 3D block found in the dictionary may not exactly match the boundary of the already reconstructed neighborhood block.

The current super-dimension reconstruction algorithm solves these problems effectively. For the error accumulation problem, dictionary optimization reduces this random error in the initial stage of reconstruction such that the reconstruction process is well controlled. For the matching error problem, the neighborhood embedding integrates multiple optimal candidate blocks and establishes an optimal mapping such that the matching error is further reduced. Therefore, the reconstruction results of the current super-dimension algorithm are closer to the CT target sample.

VI. CONCLUSION

This paper discussed the following dictionary-related issues in the field of digital core reconstruction: (1) the completeness of the pattern set (dictionary) that affects the accuracy of the reconstruction results, and (2) the one-to-many mapping problem in the dictionary, i.e., how to establish the optimal mapping relationship. These two problems are not only urgent problems to be solved in the classical multipoint geological reconstruction algorithm in the field of digital core reconstruction but also key issues in the super-dimension reconstruction recently proposed by our group. This paper discussed these common problems from the perspective of SD dictionaries.

For the completeness of the pattern set (dictionary), to expand the completeness of the dictionary, we proposed the concept of augmentation dictionaries. By rotating each patch through different orientations (0°, 90°, 180°, or 270°), the dictionary can contain more objective correspondences. To further expand the expressive ability of the dictionary, we proposed clustering based on the boundary features of the dictionary elements. First, we used gradient-based edge detection approaches to detect the boundary of a 2D image patch. The boundaries were then divided into different types by defining the "on-edge" ratio δ of the 2D image patch. Through cluster analysis, patterns with higher similarity in the dictionary can be grouped into one class, and unordered pattern sets can be organized by category. Thus, the category of the 2D patch to be reconstructed can be rapidly identified and similar patterns in this category can be found, which further improves the expressive ability of the dictionary.

For the one-to-many mapping problem in the dictionary, i.e., how to establish the optimal mapping relationship, we introduced the idea of neighborhood embedding in super-resolution reconstruction to the area of SD reconstruction. For the 2D patch to be reconstructed, when the 3D block cannot be found in the dictionary such that it completely matches the boundary of the already reconstructed neighborhood block, the optimal dictionary mapping relationship can be obtained by the neighborhood embedding method. In SD reconstruction, there is no manifold similarity between the 2D patch and the 3D block, such as that between low- and high-resolution blocks in super-resolution. Hence, it is necessary to propose neighborhood embedding suitable for super-dimension reconstruction. To solve the above-mentioned problems, by focusing on the characteristics of super-dimension, this paper proposed a constraint neighbor embedding-based dictionary mapping for SD reconstruction. For the similarity measure function, we proposed a similarity measure function that combines the 3D neighborhood information and the Markov chain information to determine the final 2D-3D matching pair. For the optimal weight, we proposed a weight determination method that combines the 3D boundary constraints.

To verify the effectiveness of the algorithm, we conducted related reconstruction experiments and analysis. Through the analysis of porosity, we found that the proposed dictionary-related algorithm can solve the problem of porosity deviation caused by an incomplete dictionary or an uncertain dictionary mapping relation in the original super-dimension. Through the analysis of the Euler number, autocorrelation function, two-point cluster function, and pore morphology parameters, we found that the proposed dictionary algorithm can improve the reconstruction accuracy by optimizing the dictionary structure and dictionary mapping relationship.

The proposed dictionary-related algorithm allows the super-dimension algorithm to better construct the dictionary structure, thereby improving the reconstruction accuracy. At the same time, it has important reference value for the pattern set problem related to the multipoint geostatistics algorithm. Of course, we must realize that the increase in dictionary completeness will inevitably affect the speed of reconstruction and require greater computer memory. How to better design the data structure and storage mode of the dictionary and how to design a better dictionary template are topics of future research.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (Grant No. 61372174).

[1] P. M. Adler, C. G. Jacquin, and J. A. Quiblier, Int. J. Multiphas. Flow **16**, 691 (1990).
[2] C. L.Y. Yeong and S. Torquato, Phys. Rev. E **58**, 224 (1998).
[3] K. M. Gerke and M. V. Karsanina, Europhys. Lett. **111**, 56002 (2015).
[4] M. V. Karsanina and K. M. Gerke, Phys. Rev. Lett. **121**, 265501 (2018).
[5] S. Strebelle, Math. Geol. **34**, 1 (2002).
[6] S. Strebelle and C. Cavelius, Math. Geosci. **46**, 171 (2014).
[7] P. Tahmasebi and M. Sahimi, Phys. Rev. E **85**, 066709 (2012).
[8] A. Hajizadeh, A. Safekordi, and F. A. Farhadpour, Adv. Water Resour. **34**, 1256 (2011).
[9] A. Comunian, P. Renard, and J. Straubhaar, Comput. Geosci. **40**, 49 (2012).
[10] H. Okabe and M. J. Blunt, Phys. Rev. E **70**, 066135 (2004).
[11] H. Okabe and M. J. Blunt, J. Petrol. Sci. Eng. **46**, 121 (2005).
[12] J. Straubhaar, P. Renard, G. Mariethoz, R. Froidevaux, and O. Besson, Math. Geosci. **43**, 305 (2011).

[13] L. Mosser, O. Dubrule, and M. J. Blunt, Phys. Rev. E **96**, 043309 (2017).
[14] J. X. Feng, Q. Z. Teng, X. H. He, and X. H. Wu, Acta. Mater. **159**, 296 (2018).
[15] Y. Li, X. H. He, Q. Z. Teng, J. X. Feng, and X. H. Wu, Phys. Rev. E **97**, 043306 (2018).
[16] Y. Li, Q. Z. Teng, X. H. He, J. X. Feng, X. H. Wu, and S. H. Xiong, J. Petrol. Sci. Eng. **174**, 968 (2019).
[17] P. Sandeep and T. Jacob, IEEE Trans. Image Process. **25**, 4233 (2016).
[18] N. Kumar and A. Sethi, IEEE Trans. Multimedia **18**, 1504 (2016).
[19] D. Mishra, B. Majhi, P. K. Sa, and R. Dash, Neurocomputing **202**, 49 (2016).
[20] W. T. Freeman, T. R. Jones, and E. C. Pasztor, IEEE Comput. Graph. **22**, 56 (2002).
[21] M. Sajjad, I. Mehmood, and S. W. Baik, J. Vis. Commun. Image R. **26**, 50 (2015).
[22] K. B. Zhang, D. C. Tao, X. B. Gao, X. L. Li, and Z. G. Xiong, IEEE Trans. Image Process. **24**, 846 (2015).
[23] X. Q. Lu, Y. Yuan, and P. K. Yan, IEEE Trans. Cybernet. **44**, 366 (2014).
[24] H. Chang, D. Y. Yeung, and Y. M. Xiong, in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR 2004, Vol. 1 (IEEE, 2004), pp. I275–I282.
[25] T. M. Chan, J. P. Zhang, J. Pu, and H. Huang, Pattern Recogn. Lett. **30**, 494 (2009).
[26] K. B. Zhang, X. B. Gao, X. L. Li, and D. C. Tao, IEEE J. Sel. Topics Signal Process. **5**, 230 (2011).
[27] J. J. Jiang, R. M. Hu, Z. Y. Wang, and Z. Han, IEEE Trans. Image Process. **23**, 4220 (2014).
[28] V. A. Rahiman and Sudhish N. George, Comput. Electr. Eng. **62**, 281 (2017).
[29] H. B. Wang, L. Feng, L. H. Yu, and J. Zhang, Neurocomputing **216**, 286 (2016).
[30] L. Feng, H. B. Wang, S. L. Liu, and H. Zhang, Int. J. Pattern Recogn. Artificial Intelligence **29**, 1551010 (2015).
[31] N. Corngold, Phys. Rev. A **6**, 1570 (1972).
[32] S. Puri and D. Kumar, Phys. Rev. E **70**, 051501 (2004).
[33] S. Torquato and H. W. Haslach, Appl. Mech. Rev. **55**, B62 (2002).
[34] H. J. Vogel, U. Weller, and S. Schlüter, Comput. Geosci. **36**, 1236 (2010).
[35] B. L. Lu and S. Torquato, Phys. Rev. A **45**, 922 (1992).
[36] S. Torquato, J. D. Beasley, and Y. C. Chiew, J. Chem. Phys. **88**, 6540 (1988).
[37] Y. Jiao and N. Chawla, J. Appl. Phys. **115**, 093511 (2014).
[38] K. M. Gerke, M. V. Karsanina, R. V. Vasilyev, and D. Mallants, Europhys. Lett. **106**, 66002 (2014).
[39] H. Dong and M. J. Blunt, Phys. Rev. E **80**, 036307 (2009).