IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 February 2023, accepted 1 March 2023, date of publication 6 March 2023, date of current version 10 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3252894

RESEARCH ARTICLE

Image Stitching Method of Bone Stick Fragment Based on Similarity Freeman Code Matching

CONGCONG LIU[1], HUIQIN WANG[1], LI MAO[1], RUI LIU[2], ZHAN WANG[3], AND TING WANG[1]

[1]School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China
[2]Institute of Archaeology, Chinese Academy of Social Sciences, Beijing 100101, China
[3]Shaanxi Institute for the Preservation of Cultural Heritage, Xi'an 710075, China

Corresponding author: Huiqin Wang (hqwang@xauat.edu.cn)

This work was supported in part by the National Social Science Fund Unpopular Special Project of China under Grant 20VJXT001.

ABSTRACT A large number of bone stick fragments were unearthed at the Weiyang Palace, and the manual stitching method is inefficient and time-consuming, therefore Image Stitching Method of Bone Stick Fragment Based on Similarity Freeman code Matching is proposed by this paper to improve bone stick stitching efficiency. The Freeman code is used to represent the contour of the true color RGB image of bone stick fragments, and the direction change dense feature of the bone stick fragment fracture is fused with the Freeman code difference to represent the change direction feature to locate the fracture boundary. At the same time, the integer basic operation is used to replace the complex calculus calculation to improve the efficiency of the algorithm; The longest common subsequence algorithm is used to perform local similarity rough matching on Freeman code of the contour at the fragment fracture, and then the angle and distance error threshold is set according to the prior knowledge of manual stitching to reduce the matching error; Finally, according to the matching results, the complete bone stick contour binary image is drawn. Compared with other algorithms, the contour Cosine similarity accuracy of the bone stick image stitched by this algorithm is improved by 21.68%, which can be accurate to 97.80%; the Editing distance similarity accuracy is improved by 14.29%, which can be accurate to 91.87%;the Peak signal-to-noise ratio is improved by 83%, which can be accurate to 31.13%;the Structural Similarity is improved by 29%, which can be accurate to 0.87; the Figure of merit is improved by 39%, which can be accurate to 0.85. The experiment results show that the algorithm proposed in this paper can effectively assist the stitching work of bone stick cultural relics.

INDEX TERMS Bone stick cultural relics, freeman code, longest common sub-sequence, roughly match, angle and distance error threshold, similarity accuracy.

I. INTRODUCTION

The bone stick cultural relics unearthed in Chang'an City of the Han Dynasty are made of animal bones and processed into long bone pieces. Bone sticks run through almost the entire Western Han Dynasty, and the written records on its surface can reflect the social, economic and political background of the Western Han Dynasty. It is of great significance to the study of Western Han culture. Due to being buried in the ground for many years, bone sticks were damaged to a certain extent when they were unearthed. Therefore, the restoration of bone stick fragments occupies a very important position in the field of archaeology. The initial stitching of fracture bone sticks can only be done manually by archaeological experts, which is extremely inefficient. With the progress of archaeological work, the number of fragments is increasing, and it is difficult to complete the stitching and repair within a limited time by manpower alone. At present, the rapid development of computer science, the application of digital image processing technology to the stitching of bone stick fragments can effectively reduce the workload and improve the stitching efficiency. Contour-based bone stick fragment stitching research can be divided into three steps: Firstly, extracting contour information of the digital image of bone

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan-Li Sun.

stick fragments; Secondly, using a appropriate matching algorithm to find fragments with the highest matching degree; Finally, restoring the matched fragments of the bone stick into a complete bone stick image. Archaeological researchers use digital cameras to convert the physical objects of the bone stick fragments into JPEG (Joint Photographic Experts Group) type digital images. After a series of analysis and experiments on the digital images of the bone stick fragments, computer technicians again stitch the images into JPEG type digital images and feed them back to archaeological researchers to assist them in completing the restoration of the bone stick relics.

There have been many researches on image matching at domestic and foreign [1], [2], [3], [4], [5], [6]. Piao et al. [7] proposed to achieve fragment matching by calculating curvature and least square fitting, extracting the surface texture features of fragments, and constructing constraints; Guo-Hua et al. [8] proposed to estimate the similarity of feature points at the fracture of fragments, and used singular value decomposition to achieve accurate matching and stitching of fragments; Jie et al. [9] proposed to construct the bidirectional distance field of the characteristic points of the fracture surface, calculate the rigid body transformation matrix, and use the iterative nearest point method to achieve the stitching of fragments; Ran-Ran et al. [10] determined the matching information of the image polygon vertices by calculating the length between the adjacent vertices of the polygon and using the improved ''tank algorithm'' to obtain the matching metric of the corner sequence. This method makes full use of the corner edge information of the outline but ignores the geometric characteristics of the curve, and the calculation amount is large when the contour representation and the vertex are matched; Gang et al. [11] proposed to select the feature points on the contour curve by calculating the curvature, and use the ratio of the arc length to the chord length of the feature segment to replace the characteristics of the curve for matching. The method can effectively improve the matching accuracy, but it requires complex curvature and arc length calculations, which will increase the execution time of the algorithm; Baozhu et al. [12] completed the matching and splicing of fragments by comparing the angle values and edge lengths of every three points on the contour. The single matching criterion of this method leads to low final stitching accuracy.

Bone stick fragments are irregular polygons. Most of the existing contour matching algorithms need to perform very complex calculations on the edge of the image, and the global contour is used as the information to be matched during matching, which leads to a long execution time of the algorithm and a large matching error. In order to solve the above problems, this paper proposes a bone stick fragment image stitching method based on similarity Freeman code [13] matching, which first locates the fracture boundary of the fragment after preprocessing operations such as denoising [14], [15] on the digital image of bone stick fragments, and then uses the LCS (Longest Common Subsequence) algorithm [16] to achieve partial rough matching, and determine the matching edge through analyzing the corner edge error, and finally reconstruct the global contour according to the matching coordinate position to achieve the purpose of restoration.

II. OUR ALGORITHM

A. SCHEME DESCRIPTION

The overall flow of the stitching method in this paper is shown in Figure 1, including three parts: image preprocessing, contour extraction and matching stitching. Firstly, the digital image of bone stick fragment is preprocessed, Freeman chain code is used to represent the outermost contour after detecting edge; Secondly, the dense features of direction change at the fracture of bone stick fragment and the error between chain codes characterize the change direction characteristic are fused to locate the fracture edge of fragments, and LCS algorithm is used to roughly match the local similarity of the Freeman code at the fracture edge of the bone sticks fragment; Then the value of Euclidean and angle between the vertices of the optimal matching edge in the rough matching result is calculated and the angle and distance error threshold is set based on the prior knowledge of multiple splicing experiments to refine the rough matching result and reduce the matching error; Finally, the original image of the bone stick is restored according to the matching results.

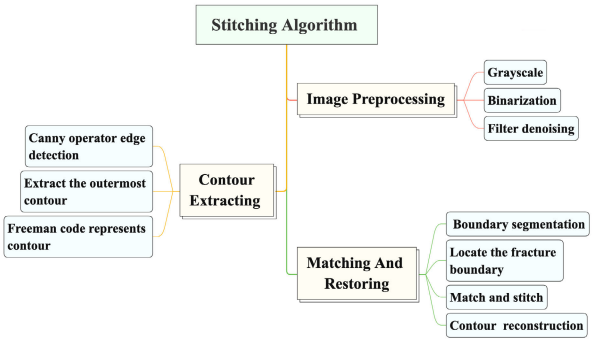


FIGURE 1. Scheme of fragment image stitching algorithm.

B. EDGE DETECTION

The original bone stick fragments are transformed into digital images by computer scanning and edge detection is required to extract the image contour after preprocessing operations such as binarization and denoising. Common edge detection operators include Robert, Sobel and the Canny, in which the Robert operator cannot smooth the image, the Sobel operator is inaccurate for edge positioning and it is easy to form a non-closed area, the Canny operator is a multi-stage optimization operator with filtering, enhancement and detection. Therefore, the Canny operator is used for edge detection to

obtain the closed boundary region of the binary image in this paper. Before processing, a Gaussian smoothing filter is defined to smooth the image to remove noise, and then the first order partial derivative finite error is used to calculate the gradient amplitude and direction. During processing, two thresholds are used to connect the edges after a non maximum suppression operation. The calculation equation is as follows:

$$H(x, y) = e^{\frac{-(x^2+y^2)}{2\times\sigma^2}} \tag{1}$$

$$g(x, y) = f(x, y) \times H(x, y) \tag{2}$$

$$G = \sqrt{G_x^2 + G_y^2} \tag{3}$$

$$\theta = \tan^{-1}(\frac{G_y}{G_x}) \tag{4}$$

where $H(x, y)$ is the system function of the Gaussian smoothing filter, (x, y) is the point coordinate, e is the natural constant, σ is the standard deviation, $g(x, y)$ is the image smoothing equation, $f(x, y)$ is the discrete Fourier transform of the image, G is the edge gradient, θ is the gradient direction, G_x and G_y are respectively the partial derivatives of the image gray level along the horizontal and vertical directions.

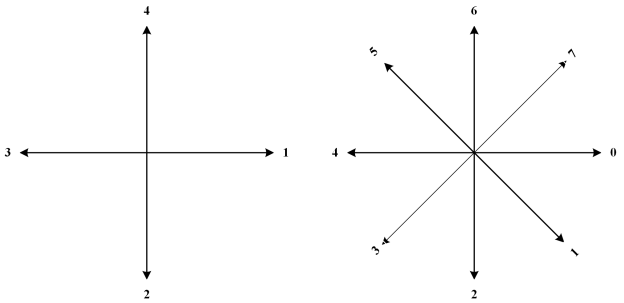
Bone sticks are processed from animal bones, and the surface is engraved with text, so the contour lines are quite complex. The Canny operator will detect all connected edges of the image. However, only the outermost contour is required for the matching operation of bone stick fragments, the internal texture information will not only increase unnecessary operations, but also interfere with the matching process, leading to errors in the matching results.

The above problem can be solved by judging that all contour areas in the edge detection results and only retaining the contour with the largest area as the outermost contour of the bone stick fragment.

C. IMPROVED FREEMAN CODE TO REPRESENT THE OUTERMOST CONTOUR

Freeman chain code is a common coding technique for expressing lines, plane curves and region boundaries, which is easier to detect the sharp corners and concave parts of the boundary. It has strong data compression ability for polygon representation. The 8-neighborhood chain code has four more bevel directions than the 4-neighborhood chain code as shown in Figure 2.

Since the edge of the bone stick fragment is a jagged line, a large amount of information will be lost if the 4-neighborhood direction chain code is used to represent the contour. Therefore, the clockwise 8-neighborhood Freeman code is used to represent the outline of bone stick fragments in this paper. The traditional chain code tracking algorithm generally tracks the edge points in the order from 0 to 7 in the 8-neighborhood directions of the Freeman code and the chain code moves and encodes along the boundary pixels in an 8-adjacent manner. The first pixel scanned is taken as the starting point of the edge chain code, and other edge points within the 8-neighborhood directions of the point are



(a) 4-neighborhood (b) 8-neighborhood

FIGURE 2. Chain code direction.

searched according to the above method, inserting the first edge point scanned in the 8-neighborhood into the chain code sequence and updating this point to the current pixel point, at the same time, marking this point to avoid repeated tracking. Then continue to search the edge points in the 8-neighborhood of the current point. Iterate repeatedly until there is no edge point in the 8-neighborhood of a current point, and finally form an edge chain code sequence.

It is too cumbersome to use the above method when there is only the outermost contour after edge detection, so this paper adopts a new chain code representation method. In the outermost contour coordinate sequence, the point with the lowest coordinate value is taken as the starting point and the counterclockwise direction (opposite to the chain code representation direction) is taken as the tracking direction of the chain code, and error is made sequentially along the edge coordinate sequence. Finally, the chain code of the contour sequence containing n pixels is expressed as $\{i, d(i), (x_i, y_i)\}^n$, (i) represents the index value of the current pixel, $i \in \{1, 2, \ldots, n\}$, the chain code value $d(i)$ represents that the current pixel i points to the next adjacent pixel $i+1$ Freeman code direction, $d(i) \in \{0, 1, 2, 3, 4, 5, 6, 7\}$, (x_i, y_i) represents the coordinate value of the current point, and the chain code value corresponding to the coordinate increment of point $i+1$ and point i is shown in Table 1.

TABLE 1. Coordinate increment and chain code value.

$(x_{i+1}\text{-}x_i, y_{i+1}\text{-}y_i)$	$d(i)$
(1,0)	0
(1,1)	1
(0,1)	2
(-1,1)	3
(-1,0)	4
(-1,-1)	5
(0,-1)	6
(1,-1)	7

Due to lighting, angle or equipment, some noise points may be mistaken as the characteristic points of the chain code change when collecting digital images, which causes errors in

locating the fracture boundary. The convex point chain code repair algorithm [17] will be used to solve this problem in this paper.

D. BOUNDARY SEGMENTATION

If the Freeman code value of a point and its adjacent pixels are not equal, it indicates that the chain code direction of the point changes, which point is the chain code feature change point. Using these feature change points to further analyze the image can simplify the image matching problem. Among the multiple boundaries of the edge contour of the bone stick fragment, the fracture boundary feature changes the most intensively, and the fracture boundary can be determined by analyzing the change of each boundary feature.

According to the principle that Freeman code 0-7 represent 8 neighborhood directions, set the chain code proportion coefficient, change threshold and direction change error, and the calculation equation is as follows:

$$f = \frac{\mathrm{M}}{8} \tag{5}$$

$$\begin{cases} Lf = SLVF \times f \\ Rf = SRVF \times f \end{cases} \tag{6}$$

$$T = \frac{\mathrm{M}}{2} \tag{7}$$

$$DVF = ||RVF - LVF| - 8| \tag{8}$$

where f is the proportion coefficient, M is the change increment of the front and back sides of the current pixel, LVF and RVF is the change factor, respectively recording the chain code values of M pixels on the front and back sides of the current pixel with the most occurrences, Lf, Rf are the direction coefficients of both sides, $SLVF$, $SRVF$ is the cumulative occurrence times of LVF and RVF respectively, T is the change threshold, DVF is the direction change error.

In combination with the error value of Freeman code indicating the change direction property, there are three cases of non-zero value of DVF, and the judgment rules of vertex are as follows:

1. $DVF \in \{1, 7\}$, the included angle of the edge composed of M pixels on both sides of the point is 135 degrees, which must be the boundary vertex;
2. $DVF \in \{2, 6\}$, the included angle of the edge composed of M pixels on both sides of the point is 90 degrees, which must be the boundary vertex;
3. $DVF \in \{3, 5\}$, the included angle of the edge composed of M pixels on both sides of the point is 45 degrees, which must not be a boundary vertex.

The scheme of boundary segmentation algorithm is shown in Figure 3:

The detailed implementation steps of the algorithm are as follows:

1) M adjacent pixels are taken respectively at the front and back sides of the current pixel;
2) Judge whether the values of LVF and RVF are equal. If they are equal, the point is not a boundary vertex because its front and rear directions are consistent, and execute 1); If not, it means that the direction of the front and rear sides of this point changes, and execute 3);
3) Calculate the direction coefficients Lf and Rf on both sides according to equation(5) and (6), and judge the relationship between them and the change threshold T according to equation(7); if both are less than or equal to T, and execute 4); otherwise execute 1);
4) Calculate the change error DVF of the current point according to equation(8). According to the vertex judgment rules above, if it is a boundary vertex, store the index value of the point in the set vertex and execute 5); if not, execute 5);
5) Add 1 to the index value of the current point to cover the current point with next pixel point, and execute 1) until all the points have been judged.
6) Traverse the vertex set. If the error between two adjacent elements does not exceed vertex distance D, calculate their three-point chain code error $Diff$ [18], and discard the element with small chain code error.

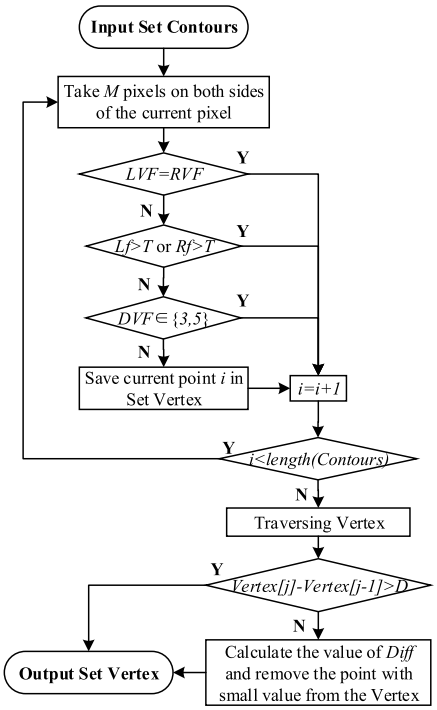


FIGURE 3. Scheme of boundary segmentation algorithm.

E. LOCATE THE FRACTURE BOUNDARY

The Freeman code of the edge contour is a discrete random variable. According to the statistical principle, under the premise that the average values are equal, the degree of data dispersion can be judged according to the standard deviation. The experimental results show that the average values of the Freeman code sequences on every boundary of the bone stick fragments are not equal, if only based on the results of the normal distribution to judge the fracture boundary of

bone stick fragments will cause errors. Another statistical index C_V(coefficient of variation) that is used to compare the degree of dispersion of sample data with different mean values can be used to solve this problem. The larger the value of C_V, the greater the error in sample data distribution. The characteristic change of the Freeman code on the boundary with the largest value of C_V of the Freeman code sequence on the bone stick fragment boundary is the most intensive, that is, the fracture boundary. The relevant calculation equation is as follows:

$$\mu = \sum_{i=1}^{n} \frac{x_i}{n} \tag{9}$$

$$\sigma = \sqrt{\sum_{i=1}^{n} (x_i - \mu)^2 \Big/ n} \tag{10}$$

$$C_v = \frac{\sigma}{\mu} \tag{11}$$

where μ is the average value of the sample data, x_i is the Freeman code value of the pixel point, σ is the standard deviation of the sample data, and C_V is the coefficient of variation of the sample data.

The vertex set stores the index values of the boundary vertices in the contour set in turn. The scheme of the algorithm for locating the fracture boundary is shown in Figure 4:

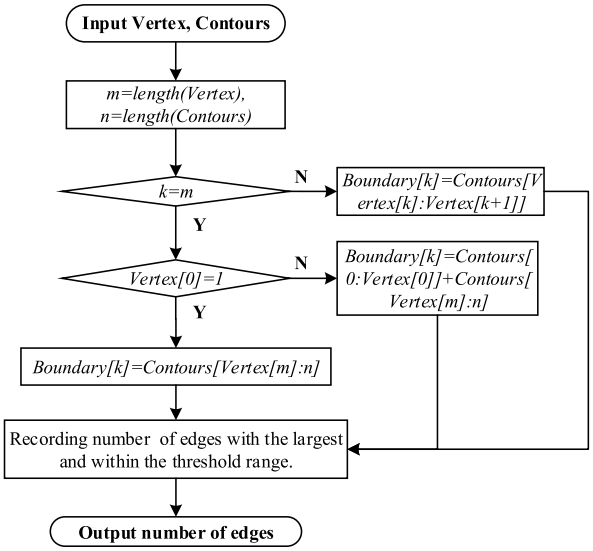


FIGURE 4. Scheme of locating the fracture boundary algorithm.

The detailed implementation steps of the algorithm are as follows:

1) When $k=\{1,2,\ldots,m\text{-}1\}$, the Freeman code sequence of the kth boundary is the Freeman code sequence between the index value of the kth vertex and the $k+1$th vertex in the entire contour;
2) When $k=m$, judge whether the value of the first element in vertex set is 1. If it is 1, the Freeman code sequence on the kth boundary is the Freeman code sequence between the mth point and the last point; if not, the first boundary vertex is not the starting point of the contour, the Freeman code sequence of the kth boundary is the Freeman code sequence between the starting point and the first vertex plus between the mth point and the last point;
3) Calculate the C_V of each boundary Freeman code sequence according to equation(9), (10) and (11);
4) The C_V of every boundary is sorted from large to small. The boundary with the largest value of C_V is the fracture boundary, and the boundary whose error is within the fracture threshold T_f is also determined as the suspected fracture boundary.

F. CORNER-DISTANCE THRESHOLD LOCAL MATCHING

After the image contour information is represented by Freeman code, the digital image information is converted into digital sequence information, and the matching problem of image contour is also converted into a digital sequence matching problem. In this paper, the LCS algorithm is combined with dynamic programming to find the best matching segment of the two sequences to be matched in each matching combination. The LCS algorithm is prone to mismatching when performing bone stick fragment boundary chain code sequence matching, so the breakpoint continuation method [11] is used to improve the fault tolerance of the matching. Calculate the Euclidean distance and the cosine of the angle between the matching boundary vertices, and set the corner-distance threshold to refine the rough matching result and reduce the matching error. Due to the matching requirements, one of the fragment contour sequences to be matched must be clockwise and the other anti-clockwise, so as to ensure the correctness of the matching. The contour chain code array in this paper is extracted in a clockwise direction, so it is necessary to reverse one of them.

The matching proportion M_P is the proportion of the best matching segment to the sequence to be matched. The matching group with the smallest error in the matching proportion in the rough matching result of LCS algorithm is the optimal result, it is preliminarily identified as a suspected matching combination. The Euclidean distance between the vertices of two boundary to be matched in the suspected matching group and the cosine value of the angle between a certain matching point and the vertex of both boundaries in the matching sequence are calculated respectively. The calculation equation is as follows:

$$M_p = \frac{LM}{LTBM} \tag{12}$$

$$M_pD = |M_{p2} - M_{p1}| \tag{13}$$

$$Ed_{a\leftrightarrow b} = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2} \tag{14}$$

$$\cos b = \frac{(\rho_{a\leftrightarrow b}^2 + \rho_{b\leftrightarrow c}^2 - \rho_{a\leftrightarrow c}^2)}{2 \times \rho_{a\leftrightarrow b} \times \rho_{b\leftrightarrow c}} \tag{15}$$

where LM is mutual matched length, $LTBM$ is the length to be matched, M_P is the matching proportion, and M_PD is the error of matching proportion, $Ed_{a\leftrightarrow b}$ is the Euclidean distance between two vertices, cos_b is the cosine value of the included

angle, (x_a, y_a), (x_c, y_c) and (x_b, y_b) are the coordinates of a, c at the vertices of both sides of the boundary to be matched and b at a matching point in the matching sequence.

The matching result is refined by comparing the error between threshold values. If the error between Euclidean distance and the error between the cosine values of the included angle are within the threshold range, it can be determined that the two sequences to be matched in this matching group are indeed matching boundaries.

The detailed implementation steps of the algorithm are as follows:

1) Extract the first fracture boundary of fragment 1 in the group to be matched as a fixed boundary;
2) Use LCS to perform rough matching between each fracture boundary of fragment 2 and the fixed boundary, and calculate the matching proportion M_P according to equation(12), which is recorded as M_{P1}, M_{P2};
3) If fragment 1 has only one fracture boundary, and execute 4); Otherwise, the other fracture boundary of fragment 1 is selected as the fixed boundary, which is roughly matched with each fracture boundary of fragment 2, until all fracture boundaries of fragment 1 and fragment 2 have completed rough matching;
4) Calculate the error of matching proportion M_PD of each matching group according to equation(13), and record the optimal result; If two fragments in the matching group have only one fracture boundary, the matching result is the optimal result by default;
5) According to equation(14) and (15), calculate the Ed (Euclidean distance) between the vertices of the two matching boundaries and the cosine of the angle between the matching points and the vertices of both sides in the optimal result;
6) If the error between Ed does not exceed the distance threshold T_d, and the error between the cosine values of the vertex angle does not exceed the angle threshold T_a, the two boundaries can be determined as matching boundaries.

The scheme of corner-distance threshold local matching algorithm is shown in Figure 5:

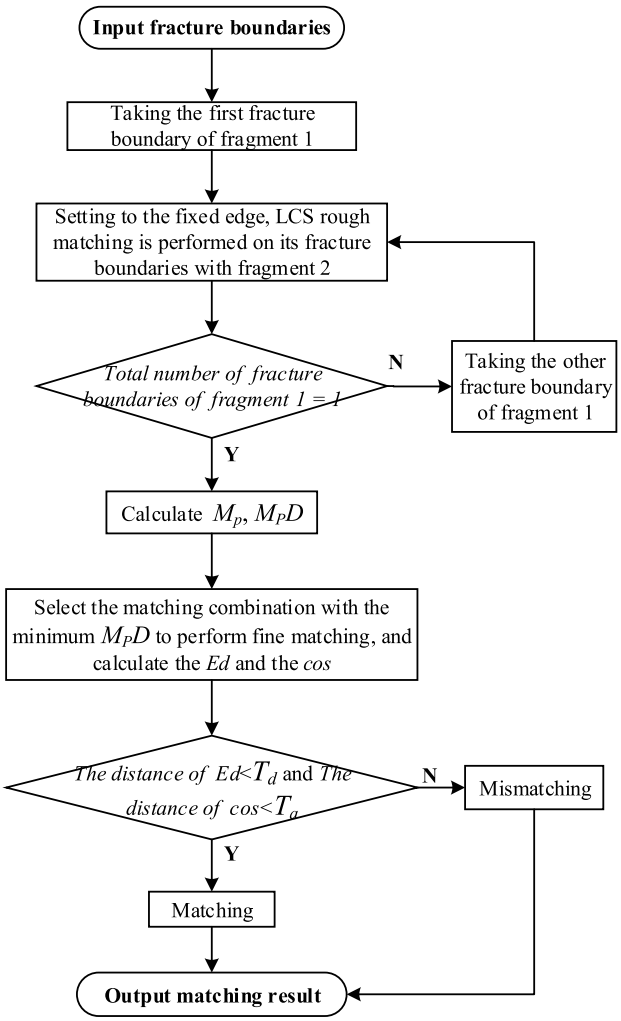


FIGURE 5. Scheme of corner-distance threshold local matching algorithm.

G. FRAGMENT STITCHING

After the matching boundary is determined, the fragment contour is restored to a complete contour image according to the matching information. The restoration steps are as follows:

1) Create a blank canvas, and reconstruct the outline of fragment 1 in the blank canvas according to the coordinate position of the original pixel point;
2) Calculate the error x and y of the horizontal and vertical coordinates of the pixels matching the fracture boundary of fragment 1 and fragment 2 in the matching result, and traverse and modify the coordinate values of the pixel points of the outline of fragment 2, the original abscissa plus x, the original ordinate plus y;
3) Reconstruct the fragment 2 outline in the canvas according to the modified pixel coordinate position.

III. EXPERIMENTAL

The acquisition hardware of the object of this study is a high-definition color camera, with backlight illumination for static shooting. The camera is installed on a fixed height detection platform, and the physical object of the bone stick is tiled on the detection platform to ensure that the complete bone stick image can be collected. The experimental PC is the Mac OS12.6 operating system, with 8 GB 2133 MHz LPDDR3 memory, 1.4 GHz quad-core Intel Core i5 processor, and Intel Iris Plus Graphics 645 1536 MB graphics card. The image processing software is programmed in the Python 3.8.2 environment.

A. ALGORITHM VERIFICATION

In order to verify the feasibility of the algorithm in this paper, four bone stick fragments unearthed from Weiyang Palace No. 3 Site in the Western Han Dynasty were divided into two

groups to be matched for matching and stitching experiments. Fragment 1 and 2 are group A to be matched, and fragment 3 and 4 are group B to be matched.

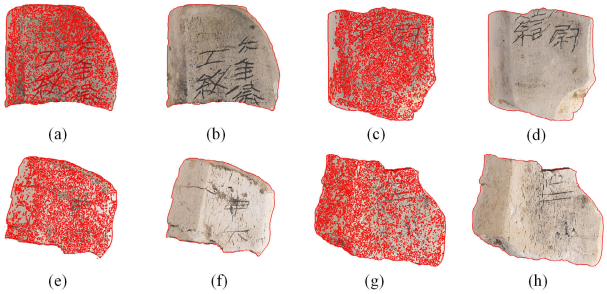


FIGURE 6. Edge contour detection image of bone stick fragments to be matched. (a) Fragment 1 edge detection (b) The outermost contour of fragment 1 (c) Fragment 2 edge detection (d) The outermost contour of fragment 2 (e) Fragment 3 edge detection (f) The outermost contour of fragment 3 (g) Fragment 4 edge detection (f) The outermost contour of fragment 4.

Figure 6 (a) and (b), (c) and (d), (e) and (f), (g) and (h) shows the result images of the edge detection and the outermost contour of the original bone stick fragment 1, fragment 2, fragment 3, and fragment 4, respectively. Set the Gaussian kernel and standard deviation of the Canny operator to 5 and 0, respectively. After edge detection of the fragment image, use the maximum area method to judge the area of all contours to locate the outermost contour.

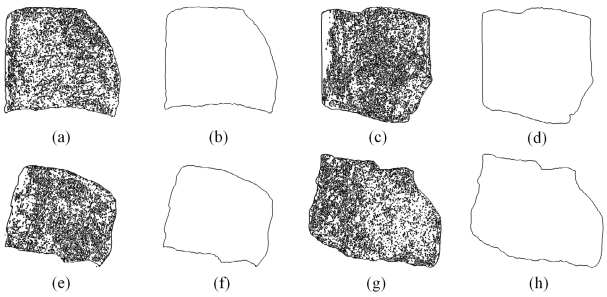


FIGURE 7. Edge contour extraction image of bone stick fragments to be matched.

Figure 7 (a) and (b), (c) and (d), (e) and (f), (g) and (h) shows the result images of the edge detection and the outermost contour extraction for reconstruction fragment 1, fragment 2, fragment 3, and fragment 4, respectively.

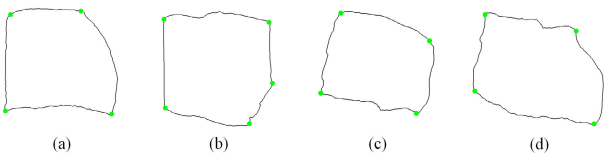


FIGURE 8. Boundary segmentation of fragment. (a) Fragment 1 (b) fragment 2 (c) fragment 3 (d) fragment 4.

Figure 8 (a), (b), (c), (d) are respectively the boundary segmentation results of fragment 1, fragment 2, fragment 3, and fragment 4 using the vertex detection algorithm, take the change increment M=10, vertex distance D = 500, 5 boundary vertices are detected in fragment 2, and 4 boundary vertices are detected in fragment 1, fragment 3, and fragment 4 respectively.

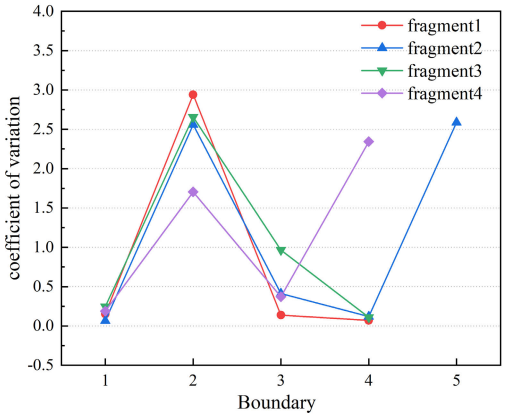


FIGURE 9. Coefficient of variation of boundary.

Figure 9 shows the C_V (coefficient of variation) result to the boundary of each fragment. Taking the fracture threshold $T_f = 0.1$, the fifth boundary of fragment 2, that is, the 2-5 C_V is the largest, and the value is 2.58707. The 2-2 C_V is 2.56028, and the error is Within the range of the threshold T_f, so 2-5 and 2-2 are both determined as fracture boundary, and the fracture boundaries of fragment 1, fragment 3 and fragment 4 are 1-2, 3-2 and 4-4 respectively, and their corresponding C_V are 2.94078, 2.65657 and 2.34277.

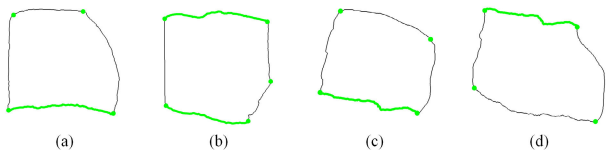


FIGURE 10. Locating the fracture boundary. (a) Fragment 1 (b) fragment 2 (c) fragment 3 (d) fragment 4.

Figure 10 (a), (b), (c) and (d) show the results of locating the fracture boundary of fragment 1, fragment 2, fragment 3 and fragment 4 respectively. The boundary with color and serial number in the figure is the detection result of locating fracture boundary algorithm.

The rough matching results of LCS algorithm are shown in Table 2. The error between the matching proportion of 1-2 and 2-2, 2-5 of the fracture boundary of matching group A is 0.190, 0.014 respectively, and the error between the matching proportion of 3-2 and 4-4 of the fracture boundary of matching group B is 0.054. According to the screening rules set by the rough matching algorithm in this paper, the minimum error between the matching proportion is the best

TABLE 2. LCS rough matching.

	group A				group B	
	1-1	2-2	1-2	2-5	3-2	4-4
LTBM	966	729	966	949	881	830
LM	675		802		778	
M_P	0.699	0.889	0.831	0.845	0.937	0.883
M_PD	0.190		0.014		0.054	

matching, that is, the error between the fracture boundary 1-2 and 2-5, 3-2 and 4-4 is the best result of matching group A and B, it is preliminarily identified as a suspected matching boundary.

TABLE 3. Angle-distance threshold fine matching.

	group A		group B	
	1-2	2-5	3-2	4-4
Ed	946.507	940.766	823.896	819.799
error	5.741		4.097	
Cos	-0.962	-0.957	-0.979	-0.956
error	0.005		0.023	

Perform the corner-distance threshold fine matching on the suspected matching boundaries, the results are shown in TABLE 3, take the distance error threshold $T_d = 10$ and the angle error threshold $T_a = 0.1$. The error s between the Euclidean distances of boundary 1-2 and 2-5, 3-2 and 4-4 are 5.74 and 4.097 respectively, and the error between the cosine values of the included angles of the matching points is 0.005 and 0.023, respectively. According to the corner-distance threshold fine matching rule, the error between the distance and the angle cosine are within the threshold. Therefore, it is determined that the fracture boundary 1-2 and 2-5, 3-2 and 4-4 are indeed matching boundary, that is, the second boundary of fragment 1 and the fifth boundary of fragment 2 match each other, the second boundary of fragment 3 and the forth boundary of Fragment 4 match each other.

Figure 11 shows the matching results of two groups of matching group fracture boundaries. The curve segments marked by the color lines are the matching segments detected by the matching algorithm in this paper. (a) and (b) are the matching results of group A fracture boundaries 1-2 and 2-2, 1-2 and 2-5, respectively, and (c) are the matching results of group B fracture boundaries 3-3 and 4-2.

Figure 12 (a), (b), (c), (d) respectively show the contour image of the bone stick reconstructed by the original method for matching group A (manual stitching by archaeologists), the local matching algorithm for locating the fracture boundary in this paper, the LCS matching method [19], and corner edge feature matching algorithm [12]. Figure 12(e), (f), (g), and (h) are the reconstructed bone stick contour image of the matching group B using the above four methods.

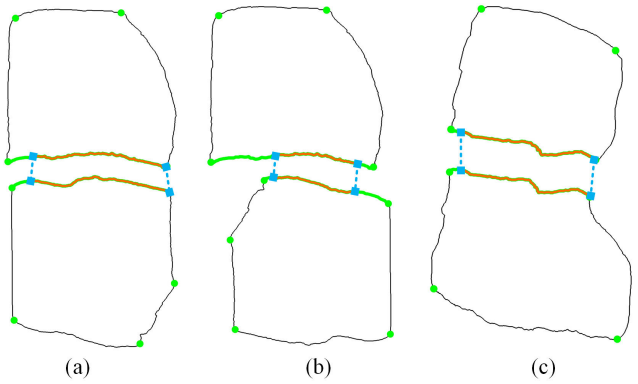


FIGURE 11. Matching results. (a) Matching group A (b) Matching group A (c) Matching group B.

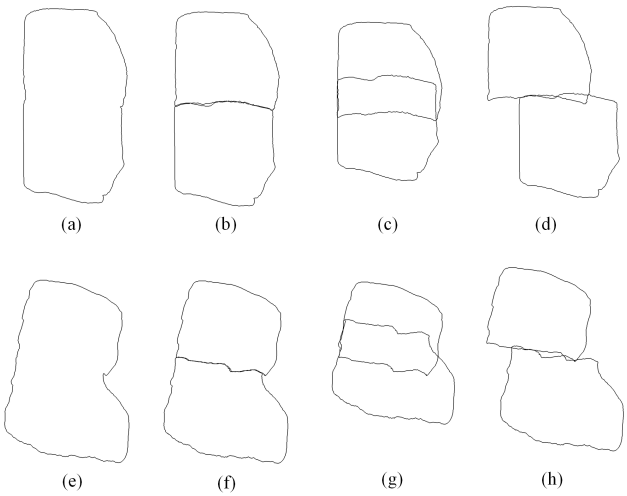


FIGURE 12. Stitching and reconstruction of bone stick contour. (a) Matching group A original method (b) Matching algorithm in this paper (c) LCS method (d) Corner edge feature matching algorithm (e) Matching group B original method (f) Matching algorithm in this paper (g) LCS method (h) Corner edge feature matching algorithm.

In this paper, CSA(Cosine similarity accuracy), EDSA (Edit Distance similarity accuracy), PSNR(Peak signal-to-noise ratio), SSIM(Structural Similarity) and FOM(Figure of merit) are used as evaluation indicators of experimental results.

CSA evaluates the similarity of two sequences by calculating the cosine value of the included angle. The closer the value of CSA is to 1, the greater the similarity between the two sequences. CSA calculation formula is as follows:

$$\mathrm{CSA} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{A_i} \times \mathrm{B_i}}{\sqrt{\sum_{i=1}^{n} (\mathrm{A_i})^2} \times \sqrt{\sum_{i=1}^{n} (\mathrm{B_i})^2}} \tag{16}$$

where $\mathrm{A_i}$ and $\mathrm{B_i}$ represent the components of sequence A and B respectively, and n represents the length of sequence.

EDSA evaluates their similarity by calculating the number of single-character edits required to convert one sequence to another. The closer the EDSA value is to 1, the greater the similarity between the two sequences. EDSA calculation formula is as follows:

$$\mathrm{EDSA} = \frac{Times}{\max(S_1, S_2)} \tag{17}$$

where *Times* represents the number of comparisons, S_1 and S_2 are two sequences to be compared.

The PSNR is used to measure the noise and distortion of the processed image. It is based on the logarithmic value of the MSE(Mean square error) of the reference image and the processed image relative to the square of the possible maximum signal value of the image. The larger the value, the smaller the noise and the higher the quality of the processed image relative to the original image. PSNR calculation formula is as follows:

$$\mathrm{PSNR} = 10 \times \log_{10}(\frac{(2^n - 1)^2}{MSE}) \tag{18}$$

where n is the number of pixel bits that determine the gray level of the image, and its value is generally 8, that is, the gray level is 256. MSE calculation formula is as follows:

$$\mathrm{MSE} = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W}(x(i,j) - y(i,j))^2}{H \times W} \tag{19}$$

where H and W respectively represent the height and width of the image; *x(i,j)* represents the coordinates of the location of the image pixel points.

SSIM evaluates the similarity between the processed image and the original image by integrating the three key features of image contrast, structure and brightness. The mean value is used as the estimate of brightness, the standard deviation is used as the estimate of contrast, and the covariance is used as the measure of structural similarity. The closer the SSIM value is to 1, the greater the similarity between the two images. SSIM calculation formula is as follows:

$$\mathrm{SSIM} = \frac{(2 \times \mu_x \times \mu_y + C_1)(\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{20}$$

where μ_x and μ_y respectively represent the average gray level of the two images, σ_x and σ_y represents the gray standard deviation of the two images, σ_{xy} represents the gray level covariance of two images, C_1 and C_2 is a constant.

FOM combines three factors: missed detection of real edge, false detection of false edge and edge positioning error. The FOM larger the value, the smaller the edge error of the two images. FOM calculation formula is as follows:

$$\mathrm{FOM} = \frac{1}{\max(S_1, S_2)\sum_{k=1}^{S_2}\frac{1}{1+\beta \times d(k)^2}} \tag{21}$$

where S_1 and S_2 are two sequences to be compared, β is the constant 1/9, and $d(k)$ is the Euclidean distance between the k-th point of two sequences.

TABLE 4. Experimental data comparison results.

		Algorithm in this paper	Algorithm [19]	Algorithm [12]
runtime /s	group A	3.042	9.258	3.280
	group B	2.877	7.849	3.055
CSA	group A	97.80%	76.50%	84.60%
	group B	96.40%	67.40%	80.40%
EDSA	group A	91.87%	76.19%	84.29%
	group B	89.51%	73.76%	78.87%
PSNR/dB	group A	30.37	30.25	29.90
	group B	31.13	30.67	30.26
SSIM	group A	0.82	0.55	0.49
	group B	0.84	0.56	0.53
FOM	group A	0.81	0.44	0.47
	group B	0.85	0.48	0.42

The experimental data comparison results are shown in Table 4. The running times of matching group A and matching group B using the matching algorithm in this paper are 3.042s and 2.877s respectively. Compared with the original restored method (manual stitching by archaeologists), the *CSA* of Freeman code sequence of the reconstructed bone stick contour by algorithm in this paper is 97.80% and 96.40%, and the *EDSA* is 91.87% and 89.51% respectively, and the PSNR is 30.37 and 31.13, and the SSIM is 0.82 and 0.84, and the FOM is 0.81 and 0.85. Compared with the LCS matching method [19], the running time of the local matching algorithm for locating the fracture boundary proposed in this paper is reduced by 5.594s on average, and the similarity accuracy of the Cosine and Editing Distance of the contour is increased by 25.15% and 15.71% on average, and the PSNR increased by 0.29 on average, and the SSIM increased by 0.28 on average, and the FOM increased by 0.37 on average. Compared with the corner and edge feature matching algorithm [12], the running time of the algorithm in this paper is reduced by 0.208s on average, and the similarity accuracy of the Cosine and Editing Distance of the contour is increased by 14.6% and 9.11% on average, and the PSNR is increased by 0.67 on average, and the SSIM is increased by 0.32 on average, and the FOM increased by 0.39 on average. The experimental results show that the matching accuracy of the proposed algorithm in this paper is higher than other algorithms.

Figure 13 (a), (b), (c), (d) and (e), (f), (g), (h) respectively show the complete images of bone stick of matching group A and B stitched and restored using the above four methods.

FIGURE 13. Restoring results of bone stick fragments. (a) Matching group A original method (b) Matching algorithm in this paper (c) LCS method (d) Corner edge feature matching algorithm (e) Matching group B original method (f) Matching algorithm in this paper (g) LCS method (h) Corner edge feature matching algorithm.

B. APPLY TO BONE STICK FRAGMENT

This paper uses the test data set named "Bone Stick Data Recovery 1" to verify the universality of the proposed algorithm in the bone stick database. The test data set contains 200 bone stick fragments, which were unearthed at the site of Weiyang Palace No. 3 in the Western Han Dynasty in February 1987.

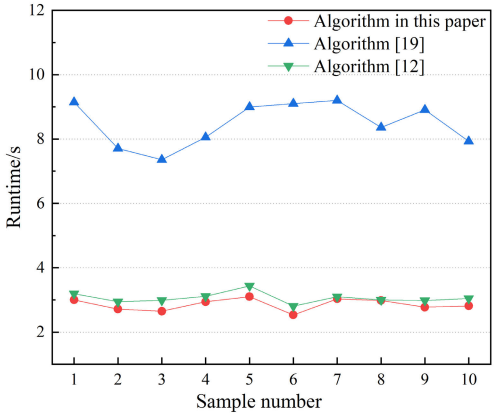


FIGURE 14. Runtime.

Figure 14 shows the runtime of different algorithms on 10 groups of bone stick data samples. The experimental results show that compared with other algorithms, the runtime of the algorithm in this paper decreases by 5.59s and 0.17s on average.

Figure 15 shows the CSA of different algorithms on 10 groups of bone stick data samples. Compared with other algorithms, the CSA of the bone stick fragment image restored by the matching algorithm in this paper and the bone stick fragment image manually stitched by archaeologists increased by 21.68% and 13.43% on average.

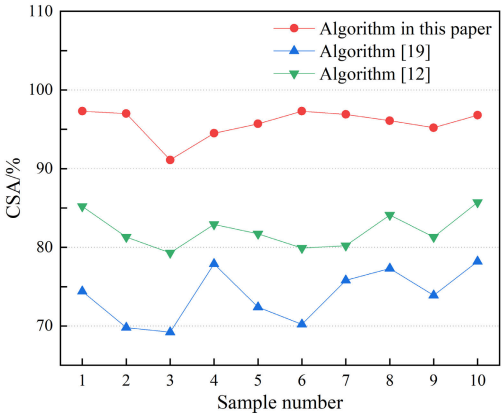


FIGURE 15. Cosine similarity accuracy.

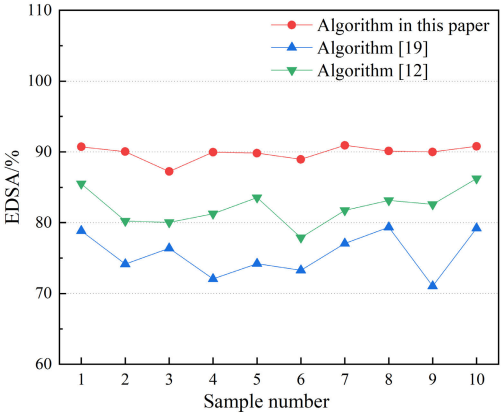


FIGURE 16. Editing distance similarity accuracy.

Figure 16 shows the EDSA of different algorithms on 10 groups of bone stick data samples. Compared with other algorithms, the EDSA of the bone stick fragment image restored by the matching algorithm in this paper and the bone stick fragment image manually stitched by archaeologists increased by 14.29% and 7.64% on average.

Figure 17 shows the PSNR of different algorithms on 10 groups of bone stick data samples. Compared with other algorithms, the PSNR of the bone stick fragment image restored by the matching algorithm in this paper and the bone stick fragment image manually stitched by archaeologists increased by 0.83 on average.

Figure 18 shows the SSIM of different algorithms on 10 groups of bone stick data samples. Compared with other algorithms, the SSIM of the bone stick fragment image restored by the matching algorithm in this paper and the bone stick fragment image manually stitched by archaeologists increased by 27% and 29% on average.

Figure 19 shows the FOM of different algorithms on 10 groups of bone stick data samples. Compared with other

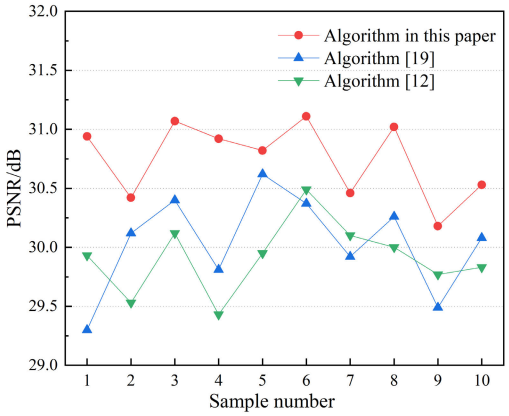


FIGURE 17. Peak signal-to-noise ratio.

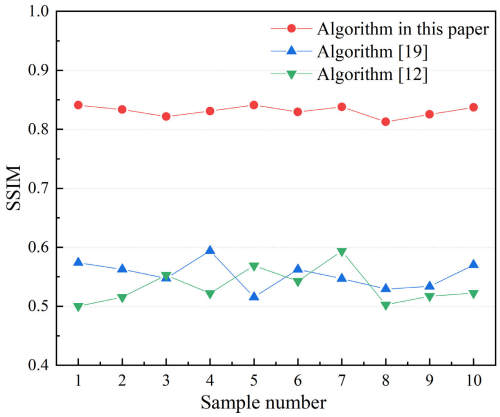


FIGURE 18. Structural similarity.

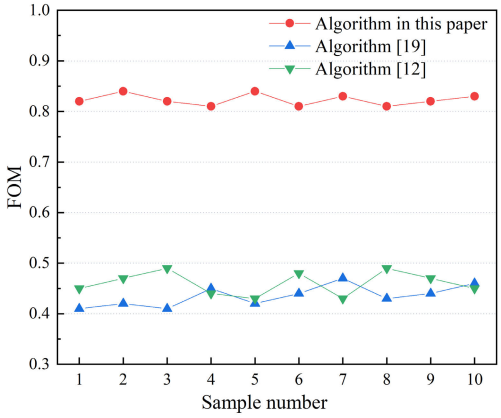


FIGURE 19. Figure of merit.

algorithms, the FOM of the bone stick fragment image restored by the matching algorithm in this paper and the bone stick fragment image manually stitched by archaeologists increased by 39% and 36% on average.

As shown in Figure 20, randomly select one fragment and other fragments to complete the rough matching of

FIGURE 20. Fragments of bone stick to be matched.

the fracture edge sequence one by one until all fragments have completed pairwise matching. Carry out corner-distance threshold fine matching for the optimal matching group in the rough matching results to determine the matching relationship, and restore the fragment that meet the matching conditions to the original bone stick, as shown in Figure 21, which result shows that the matching algorithm proposed in this paper can achieve effective stitching of bone stick fragments.

FIGURE 21. Restoring results of bone stick fragments. (a) Group 1 (b) Group 2 (c) Group 3 (d) Group 4 (e) Group 5.

IV. CONCLUSION AND FUTURE WORK

In this paper, we propose a method based on similarity Freeman code matching for the stitching of bone stick fragments, which has achieved good stitching results for the bone stick fragments with obvious fracture boundary features. However, because the matching algorithm in this paper is based on edge features, there are some bone stick fragments in the existing bone stick cultural relics database that have not obvious fracture boundary features or missing features, which will cause some errors when using the algorithm in this paper to match such data images; Therefore, our next research direction is to find a more accurate method to locate the fracture boundary and combine the texture and color features of the fragment image to complete the matching and stitching of the cultural relics of the bone stick.

REFERENCES

[1] J. S. Spaulding and G. M. Picconatto, "Characterization of fracture match associations with automated image processing," *Forensic Sci. Int.*, vol. 342, Jan. 2023, Art. no. 111519, doi: 10.1016/J.FORSCIINT.2022.111519.

[2] L. Weizhen, S. Yan, H. Xuegang, and Y. Chun, "An improved feature matching algorithm and its application in measurement for modelling of debris clouds," *J. Vib. Shock*, vol. 40, no. 2, pp. 29–38, 2021, doi: 10.13465/j.cnki.jvs.2021.02.005.

[3] L. Xiao-Ning, D. Hong-Zhang, Y. Wen, L. Peng-Yue, and W. Shi-Xiong, "Mosaic of cultural relics fragments based on SURF feature extraction descriptor and Jaccard distance," *Opt. Precis. Eng.*, vol. 28, no. 4, pp. 963–972, 2020.
[4] S. Chong, C. Ligang, F. Hao, and W. Junxiang, "Two-dimensional stitching algorithm of potsherd based on shape constraint," *J. Ceram.*, vol. 42, no. 2, pp. 320–324, 2021, doi: 10.13957/j.cnki.tcxb.2021.02.012.
[5] 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.
[6] C. Toler-Franklin, B. Brown, T. Weyrich, T. Funkhouser, and S. Rusinkiewicz, "Multi-feature matching of fresco fragments," *ACM Trans. Graph.*, vol. 29, no. 6, pp. 1–12, Dec. 2010, doi: 10.1145/1882261.1866207.
[7] W. Piao, G. Guohua, and Z. Yuhe, "Fragment splicing method based on surface texture characteristic," *Laser Optoelectron. Prog.*, vol. 55, no. 8, pp. 274–282, 2018.
[8] G. Guo-Hua, Z. Peng-Fei, L. Yu-Meng, Z. Ming-Quan, Y. Wen-Min, and L. Kang, "Reassembly method of cultural relic fragments based on the neighborhood characteristics of fracture surface," *Opt. Precis. Eng.*, vol. 29, no. 5, pp. 1169–1179, 2021.
[9] Y. Jie, Z. Mingquan, G. Guohua, and Z. Yuhe, "Heritage debris splicing algorithm based on contour line two-way distance field," *Comput. Eng.*, vol. 44, no. 6, pp. 207–212 and 218, 2018, doi: 10.19678/j.issn.1000-3428.0046520.
[10] F. Ran-Ran, H. Fa-Zhong, and X. Hua-Mei, "Local matching for 2-D fragments reassembling," *Mod. Electron. Techn.*, vol. 38, no. 9, pp. 54–56, 2015, doi: 10.16652/j.issn.1004-373x.2015.09.038.
[11] X. Gang and L. Lei, "A method for two-dimensional fragments matching based on feature points," *Sci. Technol. Eng.*, vol. 14, no. 5, pp. 129–132, 2014.
[12] S. Baozhu and L. Mei'an, "Automatic stitching and restoration algorithm for paper fragments based on angle and edge features," *J. Comput. Appl.*, vol. 39, no. 2, pp. 571–576, 2019.
[13] H. Freeman, "On the encoding of arbitrary geometric configurations," *IEEE Trans. Electron. Comput.*, vol. EC-10, no. 2, pp. 260–268, Jun. 1961, doi: 10.1109/TEC.1961.5219197.
[14] A. Khmag, A. R. Ramli, and N. Kamarudin, "Clustering-based natural image denoising using dictionary learning approach in wavelet domain," *Soft Comput.*, vol. 23, no. 17, pp. 8013–8027, Sep. 2019, doi: 10.1007/s00500-018-3438-9.
[15] A. Khmag, "Additive Gaussian noise removal based on generative adversarial network model and semi-soft thresholding approach," *Multimedia Tools Appl.*, vol. 82, no. 5, pp. 7757–7777, Feb. 2023, doi: 10.1007/s11042-022-13569-6.
[16] Z. Zijun, W. Hong, and Y. Cheng, "Two algorithms to solve longest circular common subsequence problems," *Appl. Res. Comput.*, vol. 37, no. 11, pp. 3334–3337 and 3358, 2020, doi: 10.19734/j.issn.1001-3695.2019.06.0258.
[17] L. Xiang-Hu, W. Tao, and Z. Xiao-Zhe, "Corner detection algorithm based on Freeman code analysis," *Comput. Syst. Appl.*, vol. 27, no. 4, pp. 202–208, 2018, doi: 10.15888/j.cnki.csa.006314.
[18] Y. Bo, L. Guo, T. Zhao, and X. Qian, "Curve matching method for FFC representation," *Comput. Eng. Appl.*, vol. 48, no. 4, pp. 5–8, 2012.
[19] G. Wenyue, L. Haiyan, S. Qun, Y. Anzhu, and J. Xiaolin, "A geometric similarity measure method of linear features based on longest common sequence," *J. Geomatics Sci. Technol.*, vol. 35, no. 5, pp. 518–523, 2018.

CONGCONG LIU was born in Shaanxi, China, in 1994. She is currently pursuing the master's degree with the Xi'an University of Architecture and Technology. Her research interest includes image processing.

HUIQIN WANG received the Ph.D. degree from Xi'an Jiaotong University, Xi'an, China, in 2002. She was engaged in her postdoctoral research work with Xi'an Jiaotong University, from 2002 to 2004. She is currently a Professor with the School of Information and Control Engineering, Xi'an University of Architecture and Technology. Her research interests include multimedia information security, digital image processing, information management, and information systems.

LI MAO received the master's degree from the Xi'an University of Architecture and Technology, Xi'an, China, in 2006, and the Ph.D. degree from Northwestern Polytechnical University, Xi'an, in 2014. He is currently an Associate Professor with the School of Information and Control Engineering, Xi'an University of Architecture and Technology. His research interests include digital image processing and signal and information processing.

RUI LIU, male, bachelor's degree in archaeology of Northwest University, master's degree in historical philology, and doctor's degree in ancient Chinese history of Fudan University, engaged in the research of Chinese archaeology, philology, historical geography, and ancient Chinese history. Researcher of the Institute of Archaeology of the Chinese Academy of Social Sciences, deputy director of the Han-Tang Research Office of the Institute of Archaeology of the Chinese Academy of Social Sciences, professor and doctoral supervisor of the Chinese Academy of Social Sciences.

ZHAN WANG received the master's degree from the University of Bologna, Italy, in 2008. He is currently a Researcher with the Technology Analysis and Testing Center, Shaanxi Provincial Institute of Cultural Relics Protection, Xi'an, China. His research interests include cultural relics detection analysis and researching.

TING WANG was born in China. She is currently pursuing the Ph.D. degree with Xi'an University of Architecture and Technology, Xi'an, China. She is currently a Teacher with Xi'an University of Architecture and Technology. Her research interests include image processing, pattern recognition, and their applications.