RESEARCH



Reunion helper: an edge matcher for sibling fragment identification of the Dunhuang manuscript



Yutong Zheng¹⁺, Xuelong Li¹⁺ and Yu Weng^{1*}

Abstract

The Dunhuang ancient manuscripts are an excellent and precious cultural heritage of humanity. However, due to their age, the vast majority of these treasures are damaged and fragmented. Faced with a wide range of sources and numerous fragments, the process of restoration generally involves two core elements: sibling fragments identification and fragment assembly. Currently, fragment restoration still heavily relies on manual labor. During the long practice, a consensus has been reached on the importance of edge features for not only assembly but also for identification. However, accurate extraction of edge features and their use for efficient identification requires extensive knowledge and strong memory. This is a challenge for the human brain. So that in previous studies, fragment edge features have been used for assembly validation but rarely for identification. Therefore, an edge matcher is proposed, working like a bloodhound, capable of "sniffing out" specific "flavors" in edge features and performing efficient sibling fragment identification accordingly, providing guidance when experts perform entity assembly subsequently. Firstly, the fragmented images are standardized. Secondly, traditional methods are used to compress the representation of fragment edges and obtain paired local edge images. Finally, these images are fed into the edge matcher for classification discrimination, which is a CNN-based pairwise similarity metric model proposed in this paper, introducing residual blocks and depthwise separable convolutions, and adding multi-scale convolutional layers. With the edge matcher, a complex matching problem is successfully transformed into a simple classification problem. In the absence of a standard public dataset, a Dunhuang manuscript fragment edge dataset is constructed. Experiments are conducted on that dataset, and the accuracy, precision, recall, and F1 scores of the edge matcher all exceeded 97%. The effectiveness of the edge matcher is demonstrated by comparative experiments, and the rationality of the method design is verified by ablation experiments. The method combines traditional methods and deep learning methods to creatively use the edge geometric features of fragments for sibling fragment identification in a natural rather than coded way, making full use of the computer's computational and memory capabilities. The edge matcher can significantly reduce the time and scope of searching, matching, and inferring fragments, and assist in the reconstruction of Dunhuang ancient manuscript fragments.

Keywords Dunhuang manuscripts, Edge features, Pairwise similarity metric, Convolutional neural networks

¹Yutong Zheng and Xuelong Li contributed equally to this work and should be considered co-first authors.

¹ Key Laboratory of Ethnic Language Intelligent Analysis and Security

Governance of MOE, Minzu University of China, Beijing, China

Introduction

Fragment assembly plays an essential role in several fields, such as biology [1] and forensic science [2]. Over the last few decades, notable progress has been made in the application of fragment reconstruction techniques in archaeology. Considerable advancements have been achieved in recombining various types of fragments



*Correspondence:

wengyu@muc.edu.cn

Yu Weng

© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/. The Creative Commons Public Domain Dedication waiver (http://creativeco mmons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated in a credit line to the data. including two-dimensional fragments like ancient books, oil paintings [3], murals [4, 5], and three-dimensional fragments, such as cultural relics [6-10] and damaged skeletal remains [11-13].

As the primary research object and foundation of Dunhuang Studies, Dunhuang manuscripts are mostly incomplete, with over 90% of them being fragmentary. Their appearance is shown in Fig. 1. Therefore, there is an urgent need for this technology to complete the task of joining fragments and combine a large number of fragments into larger and more complete scrolls. They are not only the treasures of Chinese cultural heritage but also a precious wealth shared by all mankind.

However, the current method of assembling Dunhuang fragments heavily relies on experts. Researchers manually piece together specific fragments based on their content, edges, and other features [14–17]. Zhang Yongguan and Luo Mujun [18] summarized 12 key factors affecting the reassemble process, such as connecting contents and edge matching, based on the characteristics of Dunhuang Buddhist sutra fragments, providing a reference for feature selection in computer-aided stitching. For past manual assembly practices, the edges serve as the most distinguishable feature. However, due to the limitations of human memory and the large number of fragments, they are primarily used to confirm whether assembly is feasible. Meanwhile, as the actual comparison is required, it poses challenges to use the edges as clues to seek out neighboring fragments.

With the development of computer technology and digital image processing, the assembly of fragments is entering the digital age. Computers have strong memory and matching capabilities to process and assemble fragments more automatically and efficiently. Therefore, It is a very effective way to use edge features for computer automatic assembly to complete the task of Dunhuang ancient manuscript fragments assembly. The essence of Dunhuang ancient manuscript fragment assembly is 2D fragment assembly by evaluating the matching probability and finding the relative position between adjacent fragments. In the procedure of 2D image composition, most of the methods exploit geometric features(such as global shape or boundaries represented by 2D curve contours) [19–27], while some focus on the content features(such as colors or patterns) [28–33]. Geometry-based pairwise matching methods rely on analyzing the shape of the boundary curve contours; color-based pairwise matching methods match fragments using their color information.

Richter et al. [34] identify pairs of corresponding points on all pairs of fragments using an SVM classifier by multimodal features of shape-and-content-based local features for aligning the respective fragments. Kang Zhang et al. [35]propose a curve-matching algorithm for automatic 2D image fragment reassembly that compute the potential matching between each pair of image fragments based on their geometry and color. Zhang et al. [36] a novel solution for the fragment assembly problem by introducing a 2D fragment assembly method that utilizes the earth mover distance to measure similarity based on length/property correspondence. Kamran et al. [37] determined the possible optimal adjacency relationship between image fragments by solving the longest common subsequence problem. Zhang Q et al. [38] proposed a contour-based 2D fragment reassembly method, which first searches for adjacent fragments in the search space and then measures the matching degree of each fragment pair through an improved polygon feature local matching method. Xin Li et al. [39]develop an image fragment descriptor called Bundle-of-Superpixel, which can more effectively support local matching and pairwise alignment.

The current algorithms rely heavily on having wellcrafted features and carefully tuned parameters.



Fig. 1 Dunhuang ancient manuscript fragments

However, this can prove to be challenging as puzzles can vary in content and complexity. Using a fixed set of handcrafted features and parameters may not be effective for all cases and parameter tuning is often difficult.

As deep learning brought efficient solutions in various computer vision tasks, we expect that the reconstruction tasks of Dunhuang ancient manuscript fragments benefit from deep learning. There is relatively little work done so far on actual the reconstruction of Dunhuang ancient manuscript fragments, possibly due to the complexity of the task and the scarcity of grounded truth samples. None of them have actually been used in the practice of the Dunhuang ancient manuscript fragments assembly.

From the above, in this paper, we focus on the approach to answering whether two fragments are from the same sheet of Dunhuang ancient manuscript or not. Specifically, we propose a method for homologous fragment identification with edges as clues. Firstly, morphological operations are performed on the fragments to extract the contours and obtain a sequence of continuous numerical type coordinates corresponding to the edge curve images of the antique fragments. Then, the Ramer-Douglas-Peucker algorithm (RDP) [40] is used to fit polygons to the continuous curves to obtain finite points, and these points are used as the center to obtain regional features near the boundary lines, which are used as local features to characterize the overall edges. The fragment pair matching task is reduced to a partial curve matching problem by connecting square regions. Finally, the local edge feature similarity of two fragments is calculated by the powerful underlying feature extraction ability of the deep convolutional neural network to realize the matching between images of ancient book fragments and complete the automatic machine assembly of ancient book fragments. In our identification method, we fully utilize edge information. Combining manually crafted features based on traditional digital image processing methods and evaluation schemes based on deep learning, not only improves the reliability and robustness of reassembly, but also enhances the interpretability and trustworthiness of the model. Experiments have shown that our method has high efficiency and accuracy, which can help people to finish the reconstruction task more quickly.

The main contributions of this paper are summarized as follows:

1. A larger-scale Dunhuang ancient manuscript fragment edge dataset, DFE-Reunion, with 36,667 images, is constructed through real Dunhuang ancient manuscript fragment chunking and manual synthesis.

- 2. An interesting idea for Dunhuang ancient manuscript fragments assembly is introduced, which converted the complex problem of matching fragment images into a simple binary classification problem of local similarity, using the validated features(edge features) of expert manual assembling as clues.
- 3. A novel CNN-based edge matcher for Dunhuang ancient manuscript fragments is proposed, which extracts local edge features and connects images by traditional digital image processing methods and designs a deep learning model to calculate the similarity between local feature image pairs. Finally, we implements AI-assisted fragment assembly as a family reunion helper for sibling fragments.

To verify the rationality of the idea and the effectiveness of the method in this paper, large-scale experiments are conducted on the benchmark dataset DFE-Reunion, comparing the matcher with recent deep learning classifiers in terms of accuracy, precision, recall, and F1-score. The recall rate reached 97.63%, demonstrating the superiority of the matcher. Our method greatly outperforms existing methods in solving the problem of ancient manuscript fragments identification.

Methods

As shown in Fig. 2, our edge matcher consists of three parts: image standardization, paired edge block region extraction, and pairwise similarity metric. The initial image standardization includes image denoising and boundary expansion, aimed at equalizing the fragment images and preventing cropping of local areas from going out of bounds. The core task is the extraction of paired edge block regions and pairwise similarity metric, which transforms the complex problem of matching fragment images into a simple binary classification problem of local edge similarities.

Given the fragments of ancient texts, we first extract the edge block areas and then connect them to obtain many candidate images. Finally, we use a CNN detector to distinguish possible correct and incorrect matches. We hope to use this as a clue to provide expert assistance and generate powerful synergies.

Image standardization

We need to standardize the input images of ancient book fragments. First, we perform Gaussian blurring to remove noise. Then, we perform precise horizontal or vertical alignment of the fragment images to ensure that the local edge areas of the two fragments are aligned as horizontally or vertically as possible. Ancient book fragments generally contain text, and the writing direction is



Fig. 2 Edge matching algorithm pipeline for Dunhuang ancient manuscript reassembly

fixed. Based on this, we can perform precise leveling, recognize the text direction through the Hough transform, and rotate the fragmented image within a certain angle range to achieve unified and automated alignment of the fragment images. We then increase the size of the original image boundaries, adding a fixed size to each of the top, bottom, left, and right sides to prevent the bounding box from exceeding the boundaries. For higher accuracy, we convert the color image to a grayscale image and then convert it to a binary image using the OTSU algorithm (Fig. 3).

In this section, there are mainly two things to be done. Firstly, precise leveling is based on the Hough transform to recognize the direction of text, rotate and correct the image, and unify and automate the orientation of the fragmented image. Specifically, a local coordinate



Fig. 3 Original image, straightened image, and binarized image

system is established, with the positive Y-axis direction (up and down) and the clockwise vertical Y-axis direction as the positive X-axis. One characteristic of ancient Chinese books is vertical writing. From bamboo slips to hand scrolls, booklets, and books, the arrangement of the text is based on the basic principle of vertical left-to-right writing. The average angle between the detected writing direction line and the positive Y-axis is calculated, and this angle is taken as the rotation angle. After obtaining the rotation angle, the image is corrected using affine transformation. Secondly, the fragmented image is binarized. After standardization is completed, all fragment images are qualified inputs for the next stage.

Paired edge block region extraction

To extract the edge block area of the fragment, there are five specific steps:

(1) To reduce the influence of internal elements on the detection edge of the operator, we use morphological operations for processing. We first erode image A with

filter B, and then subtract the result of the erosion from A to obtain $\beta(A)$. erosion can be expressed as Θ . The formula is shown as follows:

$$A\Theta B = \{z | (B)_z \subset A\},\tag{1}$$

where *A* is a set of foreground pixels, *B* is a structuring element, and the *z*'s are foreground values(1's).

$$\beta(A) = A - A\Theta B,\tag{2}$$

where $A \Theta B$ is an erosion operation.

(2) Extract the contour of the fragment by using the Canny operator, sort the contour list according to the area, and obtain the contour with the largest area, which is the edge contour of the fragment.

(3) Obtain the center point. There are two methods: (a) As shown in Fig. 4, using the RDP algorithm to approximate the boundary contour of the residual fragment as a polygon and obtain a limited number of points. As shown in Fig 5, The square area with these points as the center is the local edge feature; (b) using the boundary contour



Fig. 5 Local edge feature region image

 Table 1
 Concatenate rules

Fi	Fj	Concatenate
U	D	Yes
L	R	Yes
D	U	Yes
R	L	Yes



Fig. 6 Concatenate block image

line as the trajectory, a sliding window is performed, and the square area is taken with the boundary contour point at this time as the center, which is the local edge feature.

(4) Transform the problem of matching fragments into a partial curve matching problem, crop the local edge feature area, and classify based on the position of the region relative to the contour centroid, up or down, left or right.

The set of center points P in a local area is:

$$P = \{p_1(x_1, y_1), p_2(x_2, y_2), \dots, p_n(x_n, y_n)\},$$
(3)

Where *n* is the number of local areas of the fragment, $p_i(x_i, y_i)$ is the coordinate value of the center point of the local area, and *i* is the serial number of the center point. If the centroid coordinate of the fragment is $c(x_0, y_0)$, *Loc* represents the position category, then the formula for calculating the category of the local area is:

$$Loc(p_{i}(x_{i}, y_{i})) = \begin{cases} U, & if \quad y_{i} < y_{0} \\ D, & if \quad y_{i} > y_{0} \\ L, & if \quad x_{i} < x_{0} \\ R, & if \quad x_{i} > x_{0} \end{cases}$$
(4)

Where *U*, *D*, *L*, and *R* represent the upper edge, lower edge, left edge, and right edge of the local edge area image respectively.

(5) Concatenate blocks, up and down, left and right. If f_i and f_i represent the ith and jth fragments, and $i \neq j$, the rules are shown in Table 1.

As shown in Fig. 6, we concatenate two edge images from different fragments. This operation is beneficial for the convergence training of the model and the interpretability of the algorithm.

A CNN detector for calculating pairwise similarity metric *Main idea*

The Dunhuang manuscripts are numerous, and the situation of fragments is even more complex. How to match the fragments is a key issue.

Therefore, for the pairwise similarity metric part, we converted it into a binary classification problem by calculating the edge-matching degree of the connected blocks on the image to determine whether they match.

Network architecture design

A new convolutional neural network model is designed by combining the structure of residual blocks (RB) [41] and depthwise separable convolution(DSC) [42]. This design is based on the following observations.

To achieve higher classification accuracy and obtain global information from local regions, we need deep and complex networks. Theoretically, we can extract more high-level features and capture more internal relationships of the target.

However, as the network depth increases, training problems become more pronounced, with significant issues such as gradient vanishing and explosion. Even, the accuracy begins to saturate or even decline, known as the degradation problem of the network. Therefore, residual blocks are introduced. Furthermore, deep networks and a large number of parameters also have the side effect of slowing down model learning speed. Model compression and lightweight model design are important means to accelerate the model, thus the depthwise separable convolution is introduced.

Therefore, the combination of this structure reduces the number of parameters in the network, and the training and testing speed is significantly faster. It can reduce the model size while maintaining model performance and improving model speed.

Moreover, adaptive improvements have been made to the network structure. Parallel convolution operations have been added according to actual needs, which we call Multiple Scale Convolutional Layers(MSCL). The image is extracted for features through convolution operations of different scales and a pooling operation, and then the resulting output is combined to form the input of the next layer of the network. The convolution kernels have three shapes: vertical rectangle, horizontal rectangle, and square, which respectively extract vertical, horizontal, and common surrounding neighborhood information. From a bionic perspective, images are viewed from three perspectives and larger convolution kernels are designed to obtain larger receptive fields, extracting more global features, which are more discriminative. We do not stack convolution kernels, but each performs its calculation. The outputs of each convolution layer are concatenated

to obtain an image with more channels. Through max pooling, we obtain the most prominent features while reducing parameters and computational complexity to prevent overfitting.

Network architecture

The input of the neural network is a 224×224×3 image, which contains two square edge regions. The original input image is processed by three convolutional blocks, namely the Multiple Scale Convolutional Layer. The convolutional block (CB) applies the following modules: CB1:

- (1) Convolution of 3 filters, kernel size 7×7 with stride 2, padding (3, 3).
- (2) Batch normalization [43].
- (3) A rectified linear unit (ReLU).

CB2:

- (1) Convolution of 3 filters, kernel size 7×2 with stride 2, padding (3, 0).
- (2) Batch normalization.
- (3) A rectified linear unit (ReLU).

CB3:

- (1) Convolution of 3 filters, kernel size 2×7 with stride 2, padding (0, 3).
- (2) Batch normalization.
- (3) A rectified linear unit (ReLU).

In the above CB outputs, since the stride of all layers is 2 and SAME padding is used, the outputs of each layer have the same size but differ in in-depth control. As the outputs have the same size, they can be stacked along the depth direction to form a depth concat layer, which is then passed through max pooling and output to the residual block.

The residual block (RB(r, h)) has two parameters: the depth of input r and the depth of output h. Each residual block has the following architecture:

- (1) Convolution of h filters, kernel size 3×3 with stride 1.
- (2) Batch normalization.
- (3) A rectified linear unit (ReLU).
- (4) Convolution of h filters, kernel size 3×3 with stride 1.
- (5) A skip connection. If r = h, then directly connect the input to the block. If $r \neq h$, then apply Convolu-

tion of h filters of kernel size 3×3 with stride 1, and following batch normalization. (6) A rectified linear unit (ReLU).

The output of the residual block is passed into a depth separable convolution. The depth separable convolution (DSC(r, h)) has two parameters: the depth of input r and the depth of output h. Depthwise separable convolution mainly consists of two processes, which are depthwise convolution and pointwise convolution. As a whole, each DSC has the following architecture:

- (1) Convolution of h filters, kernel size 3×3 with stride 2
- (2) Batch normalization.
- (3) A rectified linear unit (ReLU).
- (4) Convolution of h filters, kernel size 1×1 with stride 1
- (5) Batch normalization.
- (6) A rectified linear unit (ReLU).

Finally, a fully connected layer converts the feature map to the one-hot vector (i.e. a 2×1 vector). Figure 7 illustrates the complete network architecture.

Solving data imbalance

We created a dataset consisting of pairs of squares with different edges from our training set by extracting pairededge square regions. We labeled each pair as a match or non-match based on whether they truly matched. However, the number of incorrect matches greatly outnumbered the correct ones. Essentially, the number of matching combinations is roughly equal to the square root of the total number of combinations. Thus, we used strategies for data augmentation from both the data itself and artificial construction to increase the number of matching pairs to balance the dataset.

Firstly, for the data itself, we are tolerant in two ways when selecting matching square regions. We reduced the precision of the RDP algorithm and increased the distribution of key fitting points to create more matches. We also reduced the sliding window step length along the edge contour of each matching fragment to extract more matched edge regions and create more matching pairs. We used parameters called "epsilon" to control the maximum distance between the fitting line or curve, and the "step length factor" to control the length of pixels traversed for each sliding window, indirectly controlling the number of matching pairs in the dataset and adjusting the balance of the dataset. Furthermore, Dunhuang manuscripts have undergone countless damage, resulting in severe damage, missing edges, and stains. To improve the



Fig. 7 The convolutional neural network architecture

model's ability to handle complex situations, we introduced tolerance at the data level, allowing edge square regions to not be perfectly aligned and can be partially aligned along the edge curve. We used a tolerance factor to control how much the edge alignment proportion accounted for the overall proportion of matching, indirectly controlling the number of matching pairs to solve the data imbalance issue in our experiments.

Secondly, we constructed a synthetic program to simulate local edge features. The synthetic computergenerated edge image program is controlled by three parameters: the number of turning points, the direction of the trend, and the amplitude of the curve wave. Using this generator, we could synthesize a large amount of fragment image data for training and testing. In addition, to simulate real situations, we added noise interference during the curve trend process, making the paired fragments similar but not identical, improving the model's tolerance to edge alignment to better meet the requirements of real data.

To explain the details of the method, taking the generation of a horizontally paired curve as an example, first determine the distance between the left and right endpoints on the horizontal axis. Then, starting from the left endpoint, maintain a rightward trend in the step length, randomly generating a path ending at the right endpoint to obtain the set E_1 :

$$E_1 = \{e_1(x_1, y_1), e_2(x_2, y_2), \dots e_n(x_n, y_n)\},$$
 (5)

The path composed of E_1 is divided into N segments, and K(K < N) segments are randomly selected to regenerate

a rightward-trending path, resulting in a path composed of a set E_2 that is similar to but distinct from it.

Finally, the path images represented by E1 and E2 are horizontally or vertically concatenated to obtain the synthesized image *I*:

$$I = Concatenate(E_1, E_2), \tag{6}$$

Using "epsilon," "step length factor," and "tolerance factor" can improve the balance of an imbalanced dataset. The obtained training set is still not perfectly balanced, but the two classes are in the same order of magnitude. However, by adding artificially constructed data, we can achieve a completely balanced set.

Experiments and results

Experimental environment and design

The present study's image standardization and paired patch extraction procedures are implemented in Pycharm2021, using the Python3.8 programming language. The paired similarity matching model, which is based on convolutional neural networks, is developed using PyTorch. The operating system employed is Ubuntu-18.04.1, with an Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz and Tesla P100 PCIe 16GB as the CPU and GPU, respectively.

We employ the typical Binary Cross Entropy (BCE) to supervise model training as follows:

$$BCELoss = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})), \quad (7)$$

where \hat{y} represents the predicted probability that the input pairs of edge fragments are compatible, and y

Table 2 Hyperparameters for network training

Hyperparameters	Values
Num class	2
Optimizer	Adam
Learning rate	1E-04
Num epochs	50
Pre-training	No
Batch size	32

Table 3 Data quantity statistics

Data type	Quantity
Data obtained from real data	14,467
Data obtained from synthetic data	22,200

represents the ground-truth label that the two fragments are compatible or not.

Table 2 displays the hyperparameter values selected for this experiment after thorough testing and experimentation.

Dataset construction

The benchmark dataset used in this experiment is the DFE-Reunion dataset, consisting of 36,667 images. The dataset is divided into two parts: a training set (50%) and a test set (50%).

Collection process

The dataset consists of two parts: (1) a collection of 11 groups and 31 fragments of joinable remains based on relevant professional literature on suffix remains, resulting in data obtained through image standardization in the methods section. The data mainly comes from the International Dunhuang Project (IDP) website, the Chinese Ancient Books Resource Database of the National Library, "The Dunhuang Manuscripts in the Library of the British Museum" (published by Sichuan People's Publishing House in 1990), "The Dunhuang Manuscripts in the National Library of China" (published by Beijing Library Press in 2005), and "The Dunhuang Manuscripts in the St. Petersburg Collection" (published by Shanghai Ancient Books Publishing House in 2001). (2) To address the issue of data imbalance, a total of data consisting of regular function curves and irregular random curves are constructed by computers using chip features.

Table 4 Real data category statistics

Data obtained from real data	Quantity
Positive example	1100
Negative example	13,367

Table 5 Synthetic data category statistics

Data obtained from synthetic data	Quantity
Positive example	13,900
Negative example	8300

Dataset statistics and visual analysis

After processing the raw data and simulating the program, the data distribution is as follows (Tables 3, 4, 5):

The distribution of positive and negative samples of real data is as follows:

The distribution of positive and negative samples in the synthesized data is as follows:

Organize and visualize the data as shown in Fig. 8.

Evaluation metrics

The evaluation metrics used in this study are precision, recall, F1 score, and accuracy, which are used to comprehensively evaluate the performance of the proposed algorithm.

1. The formula for calculating the precision is as follows:

$$Precision = \frac{TP}{TP + FP},\tag{8}$$

2. The formula for calculating the recall is as follows:

$$Recall = \frac{TP}{TP + FN},\tag{9}$$

3. The formula for calculating the F1 score is as follows:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall},$$
(10)

4. The formula for calculating the accuracy is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP},$$
 (11)

where *TP* is a correctly predicted positive sample, *TN* is a correctly predicted negative sample, *FP* is a negative sample incorrectly predicted as a positive sample, and *FN*



Fig. 8 Visual display of dataset statistics

 Table 6
 Comparison of experimental results on balanced or unbalanced data

Data	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Unbalanced data	19.5	95.3	19.5	27.0
Balanced data	97.630	97.668	97.630	97.638

is a positive sample incorrectly predicted as a negative sample.

The Precision of a model reflects its ability to distinguish negative samples, with higher Precision indicating a stronger ability to distinguish negative samples. The Recall reflects the model's ability to recognize positive samples, with higher Recall indicating a stronger ability to recognize positive samples. The F1 score is a combination of both, and a higher F1 score indicates a more robust model. The accuracy is used to evaluate the overall classification performance.

5. Meanwhile, considering the model performance, the running time of the model is also taken as an evaluation metric, i.e., the total time required to train the model, calculated in seconds.

Data comparison experiment

To demonstrate the effectiveness of solving the data balance problem, we designed a comparative experiment with the independent variables being the unbalanced data and the balanced data.The experimental environment, model settings, and other conditions were all the same.The experimental results are as follows:

As shown in Table 6, the balanced data evaluation metrics are all higher than those of the unbalanced data. Additionally, it's noteworthy that the precision rate of the unbalanced data is higher, which is due to the fact that the model almost always selects the category with a larger number when making predictions, leading to a high precision rate. However, this does not necessarily indicate that the model's overall performance is good, as it might not be able to accurately predict other less common categories.Furthermore, due to data imbalance, the model rarely encounters samples from rare categories during training, which might result in poor prediction capabilities for new samples from these categories in practical applications. This could lead to the inability of the model to effectively handle rare categories in realworld scenarios.

Ablation experiment

To verify the effectiveness of each part of the model and the degree of its impact on the final matching classification results, we conducted ablation experiments under different schemes while keeping other conditions fixed, including (1) the ResNet [41] basic model, (2) DenseNet [44] basic model, (3) removal of some DSC from Mobile-NetV1 [42], (4) combination of DSC and DB, (5) combination of DSC and RB, (6) combination of DSC and DB with a multi-scale convolutional layer, and (7) combination of DSC and RB with multi-scale convolutional layer. The results are shown in Table 7, where DSC, DB, RB, and MSCL represent Depthwise Separable Convolution Layer, Dense Block, Residual Block, and Multiple Scale Convolutional Layers, respectively. The performance

Algorithm scheme	Layer combinations	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
1	resnet18	95.217	95.293	95.217	95.166
2	densenet121	94.450	94.485	94.450	94.398
3	MobileNetV1	93.140	93.648	93.140	93.221
4	MobileNetV1-DSC	90.028	90.658	90.028	89.685
5	RB+DSC	96.697	96.691	96.697	96.692
6	DB+DSC	94.381	94.380	94.381	94.343
7	DB+DSC+MSCL	94.968	95.052	94.968	94.911
8	RB+DSC+MSCL	97.630	97.668	97.630	97.638

 Table 7
 Comparison of accuracy, precision, recall and F1 score of different algorithm

 Table 8
 Comparison of time consumption, and model complexity of different algorithm

Algorithm scheme	Layer combinations	Time consumption (s)	FLOPs (MMac)	Params (M)
1	resnet18	2936.007	1820	11.18
2	densenet121	6109.044	2880	6.96
3	MobileNetV1	3119.800	582.9	3.21
4	MobileNetV1-DSC	2774.932	371.35	2.13
5	RB+DSC	2508.441	994.86	11.32
6	DB+DSC	4727.087	2130	4.48
7	DB+DSC+MSCL	4798.149	2080	4.48
8	RB+DSC+MSCL	2549.019	944.28	11.33



Fig. 9 Comparison of recall rates among different algorithm schemes

differences of the models are compared in terms of accuracy, precision, recall, and F1 score, as shown in Table 7.

The performance differences of the models are compared in terms of time consumption, and model complexity, as shown in Table 8.

The comparison of recall rates achieved by different algorithm schemes is shown in Fig. 9.

Through experiments, it can be seen that the evaluation metrics of algorithm scheme 1, algorithm scheme 2, and algorithm scheme 3 have achieved accuracy, precision, recall, and F1 values of over 93%, indicating the feasibility of the idea proposed in this article of converting complex matching tasks into simple binary classification problems. The evaluation metrics of algorithm scheme 4 are slightly lower than those of algorithm scheme 3, which can indicate the necessity of the DSC module. Algorithm Scheme 5 and algorithm scheme 6 not only have improved evaluation metrics compared to algorithm scheme 1 and algorithm scheme 2, but also effectively reduce the time, demonstrating the effectiveness of combining DSC with RB or DB. Through the comparison of algorithm scheme 5 and algorithm scheme 6, the superiority of combining RB with DSC can be observed. Similarly, algorithm scheme 7 and algorithm scheme 8 can prove the effectiveness of the MSCL module compared to algorithm scheme 5 and algorithm scheme 6. Through the comparison of algorithm scheme 7 and algorithm scheme 8, the superiority of the final solution proposed in this article is proven, with accuracy, precision, recall, and F1 values significantly improved to over 97%. Overall, considering the classification metrics, time consumption, and model complexity achieved on the test set, algorithm scheme 8, which combines RB, DW, and MSCL, should be chosen to construct the model.

Comparison with other models

To demonstrate the superiority of our method, we designed a comparative experiment with reference [34]. The result is shown in Table 9.

To further test that the model proposed in this study has an obvious effect on edge matching, experiments are conducted to quantitatively and comparatively analyze eight commonly used classic convolutional neural network models as benchmark models, including AlexNet [45], VGG11 [46], ResNet18 [41], DenseNet121 [44],

Table 9 Comparison of accuracy, precision, recall, and F1 scoreamong different methods

models	Accuracy(%)	Precision(%)	Recall(%)	F1 score(%)
[34]	65.0	33.0	50.0	39.0
Ours	97.630	97.668	97.630	97.638

SqueezeNet1.0 [47], MobileNetV1 [42], MobileNetV2 [48], and MobileNetV3 [49].

When training deep learning models, to ensure that each model can receive relatively fair training, the same hyperparameters are set for each model and fixed so that they would not change during subsequent model training. The same parameter settings can ensure that all models are subject to the same constraints during training and testing, making them comparable. This can eliminate performance differences caused by different parameter settings. Each model is trained and tested under the same conditions, ensuring the fairness and credibility of the experimental results. This allows for a direct comparison of their performance indicators and an understanding of their performance advantages and disadvantages. Then, the performance of these eight models is compared with the improved model in this paper, and the comparative analysis of the evaluation indicators of these eight models is shown in Tables 10 and 11.

The recall rate compared to other convolutional neural network classification models is shown in Fig. 10.

 Table 10
 Comparison of accuracy, precision, recall, and F1 score among different models

Models	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
AlexNet	93.697	93.800	93.697	93.611
VGG11	89.163	89.886	89.163	88.749
ResNet18	95.217	95.293	95.217	95.166
DenseNet121	94.450	94.485	94.450	94.398
SqueezeNet1.0	65.066	42.335	65.066	51.295
MobileNetV1	93.140	93.648	93.140	93.221
MobileNetV2	94.401	94.518	94.401	94.326
MobileNetV3	75.243	77.788	75.243	75.764
Ours	97.630	97.668	97.630	97.638

 Table 11
 Comparison of time consumption and model

 complexity among different models

Models	Time	FLOPs (MMac)	Params (M)
	consumption (s)		
Alexnet	2131.458	711.47	57.01
Vgg11	4926.460	7630	128.77
Resnet18	2936.007	1820	11.18
Densenet121	6109.044	2880	6.96
Squeezenet1.0	2881.369	743.19	1.25
MobileNetV1	3119.800	582.9	3.21
MobileNetV2	3332.386	318.96	2.23
MobileNetV3	2378.456	58.82	1.52
Ours	2549.019	944.28	11.33



Fig. 10 Comparison of recall rates among different models

Considering the practical application scenario of ancient book patching, which is to input an image of an ancient book fragment and return a candidate patching result that matches the edge of the image. It is expected that these images contain ancient book fragment images that can truly match the input image. Therefore, the main evaluation metric in this paper is the recall rate/coverage rate. It refers to the ratio of the number of images that correctly find matching images with the chipped mouth in the actually paired matching to the total number of images returned in each image returned in patching matching candidate image.

As shown in Table 10, our algorithm is superior to the comparison algorithm in the recall evaluation metric, reaching 97.630%, especially improved by 2.413% compared to the backbone algorithm resnet18, which reflects the superiority of our model. Moreover, the highest values are also achieved in Accuracy, Precision, and F1-score. The main reason for this improvement is the addition of parallel convolution layers, which considers contour matching from different perspectives of the longitudinal neighborhood, lateral neighborhood, and surrounding neighborhood, and integrates multiple perspectives of chipped mouth features, which can better simulate human visual feature representation. In terms of training time consumption, it also has a superior level. Especially, the training time of the model is reduced by 13.18% compared to the resnet18 model, mainly because depthwise separable convolution is introduced, which reduces FLOPs by 48.11% and improves the training speed of the model.

Based on comprehensive experiments, it has been shown that the model can achieve a reduction in model complexity while maintaining model performance and improving model speed.

In addition, we can see that, except for SqueezeNet and MobileNetV3, the other classification networks have achieved good results, around 90%, which proves the effectiveness of the algorithm proposed in this paper and the feasibility of transforming matching problems into classification problems. Based on this, combining traditional image processing methods and deep learning classification models also has good interpretability.

Conclusions

This study proposes a novel way of thinking about the matching task as a classification problem. A fragments edge matcher was implemented to work as a reunion helper that can make use of edge features as identification clues naturally that haven't been achieved before. To begin with, the dataset is expanded through data synthesis and standardized. Then, the local edge feature descriptors of each fragment are constructed based on traditional digital image processing methods. Thus, the overall fragment features are characterized, and the problem transformation is completed. Finally, we create and improve a pairwise edge similarity matcher based on convolutional neural networks. Comparative and ablation experiments were subsequently conducted. The matcher achieves a recall rate of 97.630%, demonstrating rationality and effectiveness. This helper has promising practical applications.

The edge matcher is a good first step towards the final goal of our research, which is actually the local matching stage. Rather than make decisions on its own, the helper works to collaborate with experts by providing suggested identification clues, leaving the final decision-making to them. It is already capable of performing a significant sorting process on existing fragments databases, providing a list of similar fragments for a requested fragment. By leveraging the fusion of traditional digital image processing methods and cutting-edge deep learning techniques, the AI-assisted system is designed to be easy to understand and interpret, ensuring Dunhuang manuscript identification is handled with greater accuracy.

The methods put forward in this study still have some limitations. First, model efficiency relies heavily on extracting regions of paired edge blocks, and an end-toend model must be built. And more methods should be added in the experimental design part for comparison to prove the superiority of the proposed method. Secondly, there are still many factors that must be considered when identifying Dunhuang manuscript fragments, such as writing style, content, and font. These considerations should be integrated in the future to improve the comprehensiveness of fragment characterization. Finally, due to the difficulty in obtaining real data from expert restoration, there is still room for improvement in the size and coverage of the validation dataset.

Therefore, future research directions will focus on the following areas:

1. Enhancing the quality of fragment data, researching on low-quality fragment image enhancement methods, such as unsupervised denoising, super-resolution reconstruction, and low-light enhancement, to construct an end-to-end model and promote the digital protection of ancient manuscripts.

- 2. Incorporating multiple factors can assist in improving identification accuracy and efficiency, and future research can explore intelligent restoration based on multimodal fusion.
- 3. Continuously collecting and organizing data to continuously expand the size of the ancient manuscript fragment dataset.

Abbreviations

RDP	Ramer-Douglas-Peucker algorithm
RB	Residual blocks
DSC	Depthwise separable convolution
MSCL	Multiple scale convolutional layers
ResNet	Residual networks
DenseNet	Dense convolutional network
VGG	Visual geometry group network

Acknowledgements

None.

Author contributions

All the authors contributed to the current work. YTZ and XLL led the writing of the article. XLL conceptualized the main research ideas of this work, conducted the experiments, and analyzed the experimental results. YW and YTZ devised the research plan, supervised the entire process, guided the direction of this study, and provided constructive advice. XLL was responsible for all revision processes. All authors read and approved the final manuscript.

Funding None

Availability of data and materials

The datasets used or analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 2 June 2023 Accepted: 18 January 2024 Published online: 13 February 2024

References

- Marande W, Burger G. Mitochondrial dna as a genomic jigsaw puzzle. Science. 2007;318:415–415.
- Justino E, Oliveira LS, Freitas C. Reconstructing shredded documents through feature matching. Forensic Sci Int. 2006;160:140–7.
- Tsamoura E, Pitas I. Automatic color based reassembly of fragmented images and paintings. IEEE Trans Image Process. 2009;19:680–90.
- Brown BJ, Toler-Franklin C, Nehab D, Burns M, Dobkin D, Vlachopoulos A, Doumas C, Rusinkiewicz S, Weyrich T. A system for high-volume acquisition and matching of fresco fragments: reassembling theran wall paintings. ACM Trans Graph (TOG). 2008;27:1–9.
- Brown B, Laken L, Dutré P, Van Gool L, Rusinkiewicz S, Weyrich T. Tools for virtual reassembly of fresco fragments. Int J Herit Digit Era. 2012;1(2):313–29.

- Jo YH, Hong S, Jo SY, Kwon YM. Noncontact restoration of missing parts of stone buddha statue based on three-dimensional virtual modeling and assembly simulation. Herit Sci. 2020;8:1–12.
- Son K, Almeida EB, Cooper DB. Axially symmetric 3d pots configuration system using axis of symmetry and break curve. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2013;pp 257–264.
- Dellepiane M, Niccolucci F, Serna SP, Rushmeier H, Van Gool L, et al. Reassembling thin artifacts of unknown geometry. In: Proceedings of the 12th international conference on virtual reality, archaeology and cultural heritage. 2011. pp. 51–61. Citeseer.
- Kashihara K. Three-dimensional reconstruction of artifacts based on a hybrid genetic algorithm. In: 2012 IEEE international conference on systems, man, and cybernetics (SMC), 2012; pp900–905
- Cohen F, Liu Z, Ezgi T. Virtual reconstruction of archeological vessels using expert priors and intrinsic differential geometry information. Comput Graph. 2013;37(1–2):41–53.
- Zhang K, Yu W, Manhein M, Waggenspack W, Li X. 3D fragment reassembly using integrated template guidance and fracture-region matching. In: Proceedings of the IEEE International Conference on Computer Vision, 2015; pp. 2138–2146.
- Yin Z, Wei L, Li X, Manhein M. An automatic assembly and completion framework for fragmented skulls. In: 2011 International Conference on Computer Vision, 2011;pp. 2532–2539.
- Wang T, Wang H, Wang K, Yang Z. Research on image mosaic method based on fracture edge contour of bone tag. Appl Sci. 2023;13(2):756.
- P CL. Piecing together and research of the Dunhuang manuscript datang tianxia jun xingshi zupu, focusing on s.5861. Dunhuang Res. 2014;01:78–86.
- X JS, L C. reconstruction of the mahāparinirvāna-sūtra from dunhuang manuscripts in the british library. Dunhuang Res. 2017;3:92–107.
- Zhang Y. The patching-up and study on the fragments of xinpusajing, quanshanjing and jiuzhuzhongshengkunanjing in dunhuang manuscripts. Fudan J (Soc Sci). 2015;57(6):12–20.
- Zhang Y, X ZR. A study on the patching-up of the fragments of golden light sutra in the Russian collections of dunhuang manuscripts. Fudan J (Soc Sci). 2015;57(6):1–1120.
- Zhang Y, Luo M. Key factors of patching up fragmentary Dunhuang Buddhist scriptures. J Zhejiang Univ (Hum Soc Sci). 2016;46(3):5–20.
- Wolfson HJ. On curve matching. IEEE Trans Pattern Anal Mach Intell. 1990;12(5):483–9.
- Kong W, Kimia BB. On solving 2d and 3d puzzles using curve matching. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, vol. 2.
- da Gama Leitao HC, Stolfi J. A multiscale method for the reassembly of two-dimensional fragmented objects. IEEE Trans Pattern Anal Mach Intell. 2002;24(9):1239–51.
- McBride JC, Kimia BB. Archaeological fragment reconstruction using curve-matching. In: 2003 Conference on Computer Vision and Pattern Recognition Workshop, 2003;1:3–3.
- Huang Q-X, Flöry S, Gelfand N, Hofer M, Pottmann H. Reassembling fractured objects by geometric matching. In: ACM SIGGRAPH 2006 Papers, 2006; pp. 569–578.
- Zhu L, Zhou Z, Zhang J, Hu D. A partial curve matching method for automatic reassembly of 2d fragments. In: Intelligent computing in signal processing and pattern recognition: international conference on intelligent computing, ICIC 2006 Kunming, China, August 16–19, 2006. Springer; 2006. pp. 645–650.
- Zhu L, Zhou Z, Hu D. Globally consistent reconstruction of ripped-up documents. IEEE Trans Pattern Anal Mach Intell. 2007;30(1):1–13.
- Li H, Zheng Y, Zhang S, Cheng J. Solving a special type of jigsaw puzzles: banknote reconstruction from a large number of fragments. IEEE Trans Multimed. 2013;16(2):571–8.
- Son K, Hays J, Cooper DB, et al. Solving small-piece jigsaw puzzles by growing consensus. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016; pp. 1193–1201.
- Amigoni F, Gazzani S, Podico S. A method for reassembling fragments in image reconstruction. In: Proceedings 2003 international conference on image processing (Cat. No. 03CH37429), 2003;3:581.

- 29. Sagiroglu MS, Erçil A. A texture based matching approach for automated assembly of puzzles. In: 18th International conference on pattern recognition (ICPR'06),2006;3:1036–1041.
- Gallagher AC. Jigsaw puzzles with pieces of unknown orientation. In: 2012 IEEE Conference on computer vision and pattern recognition, 2012;382–389.
- Sholomon D, David O, Netanyahu NS. A genetic algorithm-based solver for very large jigsaw puzzles. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2013;1767–1774.
- 32. Paikin G, Tal A. Solving multiple square jigsaw puzzles with missing pieces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2015;4832–4839.
- Gur S, Ben-Shahar O. From square pieces to brick walls: the next challenge in solving jigsaw puzzles. In: Proceedings of the IEEE international conference on computer vision, 2017;4029–4037.
- Richter F, Ries CX, Cebron N, Lienhart R. Learning to reassemble shredded documents. IEEE Trans Multimed. 2012;15(3):582–93.
- Zhang K, Li X. A graph-based optimization algorithm for fragmented image reassembly. Graph Models. 2014;76(5):484–95.
- Zhang M, Chen S, Shu Z, Xin S-Q, Zhao J, Jin G, Zhang R, Beyerer J. Fast algorithm for 2d fragment assembly based on partial emd. Vis Comput. 2017;33:1601–12.
- Kamran H, Zhang K, Li M, Li X. An lcs-based 2d fragmented image reassembly algorithm. In: 2018 13th International conference on computer science & education (ICCSE), 2018; pp. 1–6.
- Zhang Q, Li L, Fang R, Xin H. Reassembly of two-dimensional irregular fragments by improved polygon feature matching. In: 2019 IEEE 11th international conference on advanced infocomm technology (ICAIT), 2019; pp. 82–85.
- Li X, Xie K, Hong W, Liu C. Hierarchical fragmented image reassembly using a bundle-of-superpixel representation. Comput Aided Geom Des. 2019;71:220–30.
- 40. Ramer U. An iterative procedure for the polygonal approximation of plane curves. Comput Graph Image Process. 1972;1(3):244–56.
- He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2016;770–778
- Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 2017.
- Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning, 2015; pp. 448–456. pmlr.
- Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017;pp. 4700–4708.
- 45. Krizhevsky A, Sutskever I, Hinton G. Imagenet classification with deep convolutional neural networks, 2012;pp. 1097–1105.
- Simonyan K, Zisserman A. Very deep convolutional networks for largescale image recognition. arXiv preprint arXiv:1409.1556 2014.
- Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. arXiv preprint arXiv:1602.07360 2016.
- Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018; pp. 4510–4520.
- Howard A, Sandler M, Chu G, Chen L-C, Chen B, Tan M, Wang W, Zhu Y, Pang R, Vasudevan V, et al. Searching for mobilenetv3. In: Proceedings of the IEEE/CVF international conference on computer vision, 2019; pp. 1314–1324.

Publisher's Note

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