SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

September 2019, Vol. 62 192102:1–192102:14
https://doi.org/10.1007/s11432-018-9724-7

Graph-matching-based character recognition for Chinese seal images

Bin SUN[1], Shaojun HUA[1], Shutao LI[1*] & Jun SUN[2]

[1]*College of Electrical and Information Engineering, Hunan University, Changsha* 410082*, China;*
[2]*Information Department, Fujitsu Research and Develop Center, Beijing* 100027*, China*

Received 5 April 2018/Accepted 31 October 2018/Published online 30 July 2019

Abstract Recognizing characters in Chinese seal images is important when researching ancient cultural artworks because the seals may contain critical historical information. However, owing to large intraclass variance and a limited number of training samples, recognizing such characters in Chinese seals is challenging. Thus, this study proposes a graph-matching-based method to recognize characters in historical Chinese seal images. In the proposed method, a Chinese seal character is first modeled as a graph representing its underlying geometric structure. Then, two affinity matrices that measure the similarity of nodes and edge pairs are calculated with their local features. Finally, a correspondence matrix is calculated using a graph matching algorithm and the most similar reference is selected as the recognition result. Compared with several existing classification methods for seal image recognition, the proposed graph-matching-based method achieves better results, particularly in the case of limited samples.

Keywords graph model, Chinese seal, character recognition, graph matching

Citation Sun B, Hua S J, Li S T, et al. Graph-matching-based character recognition for Chinese seal images. Sci China Inf Sci, 2019, 62(9): 192102, https://doi.org/10.1007/s11432-018-9724-7

1 Introduction

In Chinese culture, seals are inextricably related to artwork, such as calligraphy and painting. To demonstrate ownership, collectors used to place their seal on ancient books, calligraphies, paintings, and other cultural relics. Such seals can provide important clues about the authenticity and heritage of cultural artwork. Therefore, being able to recognize the characters in the seals is important for historical and cultural studies. Differing from conventional character recognition, seal character recognition can be challenging. Compared with Chinese text characters, the number of seal character samples is much smaller; thus, sufficient training samples may not be available. As shown in Figure 1(a), a seal image contains only four characters. In addition, significant intraclass variance can occur because the same character may have a variety of forms. Thus, a method to determine the correspondence between two seal characters is required. Figure 1(b) shows six forms of the same character “yin”(seal). The limited number of samples and variations among the form of characters markedly increase the difficulty of establishing correspondence between seal characters.

In the last few years, most seal detection and recognition studies have focused on modern seals [1], such as administrative [2] and financial seals [3], where characters are generally presented using standardized fonts. However, few studies have investigated character recognition of seals on Chinese paintings. Related fields, such as handwritten character recognition [4, 5] and sketch recognition [6], have made significant

* Corresponding author (email: shutao_li@hnu.edu.cn)

(a) (b)

Figure 1 (Color online) Examples of ancient Chinese seals. (a) Ancient personal seal; (b) different forms of the character "yin" (seal) in seal fonts.

progress in the past few years. This progress has depended on deep models, the availability of a large number of labeled samples, and efficient computation. However, faced with the major challenges inherent in the recognition of ancient seal characters, we attempt to solve this problem from a different perspective. Rather than the color or thickness of a stroke, we believe that the geometric structure is the most intrinsic feature of a seal character. Therefore, in the proposed method, we represent the geometric structure information of the seal character using a graph model such that the character can be recognized via graph matching using only a small number of labeled samples.

Graph matching is a technique used to solve the correspondence problem between two graphs or point sets. Graph matching has been applied to shape matching [7–9], object categorization [10, 11], feature tracking [12], and action recognition [13, 14]. Various graph matching methods have been proposed. For example, Belongie et al. [15] proposed a simple but efficient shape matching method using shape contexts that captures the distribution properties of points in a specified local region for each key point. Liu et al. [16] proposed a heuristic search approach to match the strokes of a target Chinese character to the corresponding strokes of a reference character. Although their study is inspiring, the performance of their method is unsatisfactory because important global structural information and local details are discarded.

Traditional methods to determine the correspondence between two pointsets include the random sampling consistency (RANSAC) method [17] and the iterative closest point (ICP) method [18]. These methods assume a transformational relation, e.g., affine transformation, between two point sets. The RANSAC method obtains matching point pairs via multiple random sampling and removes outliers. Then, transformation parameters are estimated using all matching pairs. The ICP method continuously selects the point nearest the target point as the corresponding point and gradually updates the transform parameters. Although these methods can effectively identify consistent correspondence between feature points, they are limited to correspondence problems with rigid transformations. The traditional methods only find correspondence for point sets. However, the geometric structure of a character contains edges between key points.

Compared to traditional methods, a graph model comprising nodes and edges (e.g., length and the orientation of edges) can better represent characters, and the correspondence between two graphs is a better measure of character similarity. Mathematically, graph matching is formulated as a quadratic assignment problem (QAP), which has been proven to be NP-hard. Therefore, researchers have primarily focused on developing efficient algorithms. For example, Cho et al. [19] proposed reweighted random walks for graph matching to solve a QAP approximately. Mateus et al. [20] proposed articulated shape matching using Laplacian eigenfunctions and unsupervised point registration. In this study, we adopt the formulism proposed by Zhou et al. [21] where an affinity matrix between two graphs is factorized as an affinity matrix of the nodes and edges. This considerably reduces the computational complexity and resource requirements of the graph matching problem.

We proposed graph-matching-based Chinese seal character recognition (GMCSCR). Our primary contributions are summarized as follows: (1) the geometrical structure of characters is extracted using a skeleton-based operation and represented using graph models, and (2) graph matching with node and edge features is utilized to compute the geometrical similarity between an input character image and a reference image for character recognition.

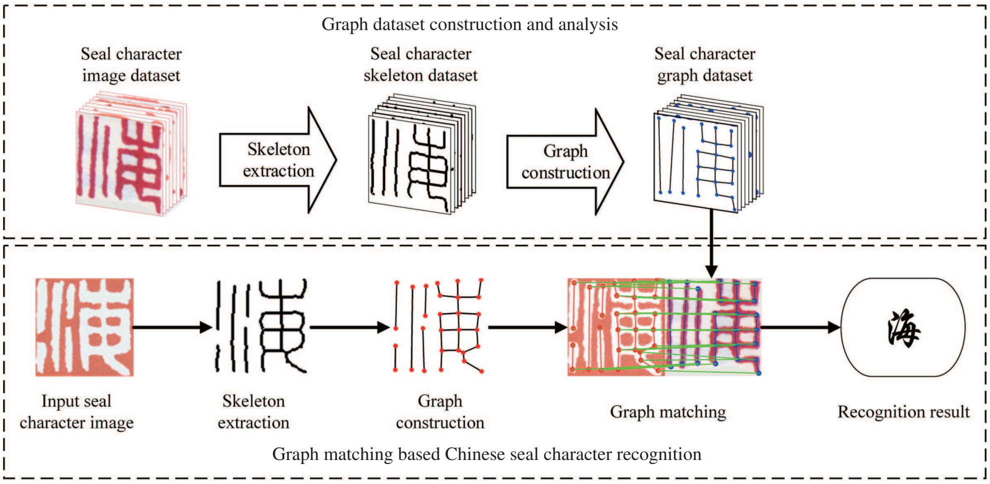

Figure 2 (Color online) Graph-matching-based character recognition for Chinese seal images.

The remainder of this paper is organized as follows. The proposed method is presented in Section 2. Experimental results are presented and analyzed in Section 3, and conclusion is provided in Section 4.

2 Proposed method

The proposed character recognition method involves graph model construction, affinity calculation, and correspondence estimation. These processes are summarized as follows.

(1) **Graph construction.** Extract a skeleton for each character and prune it to obtain the key points and connectivity relations of these points.

(2) **Affinity matrix calculation.** Calculate the node and edge affinity matrices with spatial location, shape context, and orientation features.

(3) **Correspondence estimation.** Optimize the graph affinity function and estimate the correspondence between the graph models of two characters in consideration of node affinity and edge affinity.

Figure 2 shows an overview of the proposed GMCSCR method. First, the geometric structure of the character is extracted and represented as a graph model. This involves a series of preprocessing steps to exclude unnecessary information, such as background, color, and stroke thickness information. In the construction of a graph model, polygonal approximation is employed to detect nodes, i.e., the key points of the skeleton, including endpoints, turning points, and branch points. Here, strokes linking two key points are the edges of the graph model. Then, the node affinity matrix $\boldsymbol{K}_v$ is calculated using features related to spatial location, shape context, and the number of connections. The edge affinity matrix $\boldsymbol{K}_e$ is calculated using spatial position, length, and edge orientation features. Finally, the correspondence matrix is calculated using a graph matching algorithm and the most similar reference character is selected as the recognition result.

2.1 Graph construction

A skeleton is widely considered to be a robust and discriminative shape descriptor. Figure 3 shows the workflow of the graph model construction of the seal character. Here, the bilinear interpolation algorithm is initially used to resample the input image $\boldsymbol{I}$ to a normalized image $\boldsymbol{I}^{\rm re}$ of fixed size. Then, to obtain binary image $\boldsymbol{I}^{\rm bw}$, the Otsu's thresholding algorithm [22] is employed to separate the character from the image background. The skeleton $\boldsymbol{I}^{\rm sk}$ is then extracted from the binarized character. To improve the visual representation, the skeleton and pruned skeleton images are displayed as negative images, as shown in Figures 3 and 4. In the skeleton of the seal character, a series of steps, e.g., locating key points, path searching, and polygonal approximation, are used to extract the key skeleton points, which are critical to the geometric structure of the seal character.

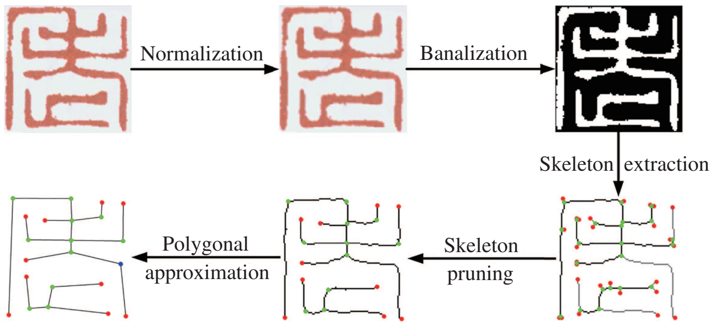

Figure 3 (Color online) Graph construction process.

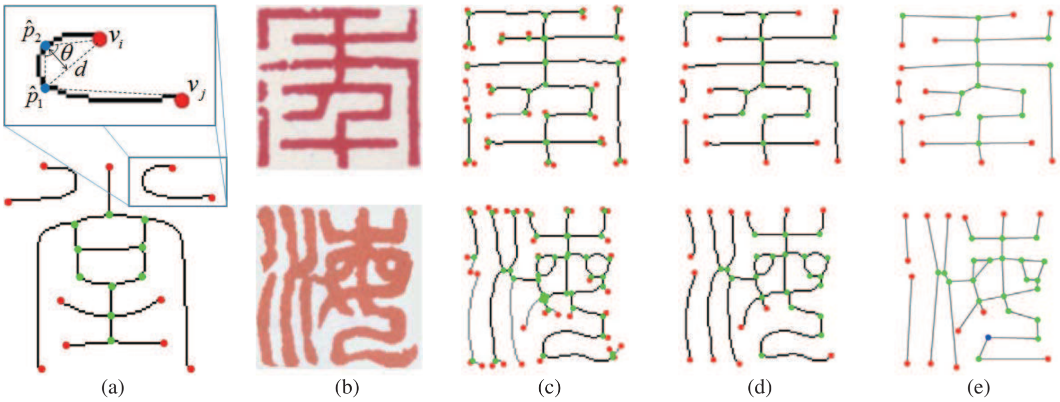

Figure 4 (Color online) Examples of graph construction result for Chinese seal character. (a) Missing turning points detected by polygonal approximation; (b) original character image; (c) skeleton extraction result; (d) skeleton pruning result; (e) graph model (endpoints, branch points, and new turning points calculated via polygonal approximation are represented by red, green, and blue, respectively).

A skeleton image $\boldsymbol{I}^{\mathrm{sk}}$ obtained from a Chinese seal character often has many small branches, as shown in Figure 4(c), and these branches, which are not a character stroke feature, adversely affect recognition accuracy. To extract the key points in the skeleton, the number of skeleton pixels in the neighborhood of each skeleton pixel is computed as follows:

$$C_i = \sum_{j \in \Omega_i} (I_j^{\mathrm{sk}}), \tag{1}$$

where Ω_i is the 3×3 neighborhood of the i-th point p_i in the skeleton, and C_i is number of skeleton pixels in the neighborhood of the i-th skeleton pixel. According to C_i, the end and branch points of the skeleton are detected as follows:

$$p_i \in \begin{cases} V_{\mathrm{end}}, & \text{if } C_i = 2, \\ V_{\mathrm{branch}}, & \text{if } C_i = 3. \end{cases} \tag{2}$$

To remove redundant branches, the path from each endpoint to the first branch point in the skeleton image is found and redundant branches whose length is less than the stroke width $l_{\min}$ and the corresponding endpoint are deleted from the skeleton. Figure 4(d) shows results of skeleton pruning Figure 4(b).

After removing redundant branches, the initial key point set of the skeleton comprises all endpoints and branch points. To obtain the neglected turning points of some strokes, polygonal approximation [23] is used to describe the path between all pairs of linked key points. Here, for each path, an iterative partition strategy is employed to detect new key points in the path. With a slight abuse of definition, we define the path point angle as the angle between the two rays from the path point to the two endpoints of the path, and the path point distance is defined as the Euclidean distance from the path point to the line of the two endpoints. The path point angle and path point distance are calculated as follows:

$$\theta(p_k) = \arccos\frac{a^2+b^2-c^2}{2ab}, \tag{3}$$

$$d(p_k) = \sqrt{\frac{(l-a)(l-b)(l-c)(a+b+c)}{c}}, \tag{4}$$

where $p_k \in \psi_{v_i\to v_j}$, $a = \|v_i - p_k\|_2, b = \|v_j - p_k\|_2$, $c = \|v_i - v_j\|_2$ and $l = (a+b+c)/2$. As shown in Figure 4(a), the polygonal approximation algorithm continues to add turning points of the path to the key point set recursively. Algorithm 1 summarizes the graph construction workflow.

Algorithm 1 Graph construction of Chinese seal character

Input: Input image $\boldsymbol{I}$; skeleton pruning threshold $l_{\min}$; polygonal approximation threshold $\theta_{\min}$, $d_{\max}$.
Output: The graph model $\boldsymbol{G} = \{\boldsymbol{V}, \boldsymbol{E}\}$.

Resize the input image $\boldsymbol{I}$ to a normalized image $\boldsymbol{I}^{\rm re}$ in fixed size 100×100;
Obtain the binary image with Otsu's algorithm: $\boldsymbol{I}^{\rm bw} = \mathrm{Otsu}(\boldsymbol{I}^{\rm re})$;
Extract the skeleton image $\boldsymbol{I}^{\rm sk}$ from $\boldsymbol{I}^{\rm bw}$;
for each skeleton point $p_i \in \boldsymbol{I}^{\rm sk} = 1$ **do**
  Find $\boldsymbol{V}_{\rm branch}$ and $\boldsymbol{V}_{\rm end}$ with (2);
end for
for each end point $v_i \in \boldsymbol{V}_{\rm end}$ **do**
  Find the path $\psi_{i\to j}$ for v_i to the first branch point $v_j \in \boldsymbol{V}_{\rm branch}$ in the skeleton images;
  if $\mathrm{length}(\psi_{i\to j}) < l_{\min}$ **then**
    Delete v_i and $\psi_{i\to j}$ from $\boldsymbol{V}_{\rm end}$ and $\boldsymbol{I}_{\rm sk}$ respectively;
  end if
  Let $\boldsymbol{V} = \{\boldsymbol{V}_{\rm branch}; \boldsymbol{V}_{\rm end}\}$;
end for
for each end point $v_i \in \boldsymbol{V}$ **do**
  if $\exists\psi_{i\to j}$ in $\boldsymbol{I}^{\rm sk}$ **then**
    $\boldsymbol{E} = \{\boldsymbol{E}; (i,j)\}$;
  end if
end for
for each edge $e_c = (i,j) \in \boldsymbol{E}$ **do**
  $n = \mathrm{size}(\boldsymbol{V})$;
  Find candidate key point $\hat{p}$ with (3);
  if $d(\hat{p}) > d_{\max}$ **then**
    Calculate the path point angle $\theta(\hat{p})$ with (4);
    if $\theta(\hat{p}) < \theta_{\min}$ **then**
      $\boldsymbol{V} = \{\boldsymbol{V}; v_{n+1} = \hat{p}\}$;
      $\boldsymbol{E} = \{\boldsymbol{E}; (i, n+1); (n+1, j)\}$;
    end if
  end if
end for
The graph model is integrated with $\boldsymbol{V}$ and $\boldsymbol{E}$: $\boldsymbol{G} = [\boldsymbol{V}, \boldsymbol{E}]$;
Return $\boldsymbol{G}$.

Figure 4(e) shows the graph model constructed from the original seal character image in Figure 4(b). As can be seen, the geometric structure of the seal character is effectively extracted and represented by the graph model. The proposed algorithm can generate graph models with a minimal number of nodes and can well preserve the geometric structure of the character strokes.

2.2 Affinity matrix calculation

Using the graph model of the seal character, we can obtain the set of key points denoted $\boldsymbol{V} = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^{n\times2}$ and the set of pairwise connections as edges $\boldsymbol{E} = [e_1; e_2; \ldots; e_m] \in \mathbb{R}^{m\times2}$. Here, $v_i = (x_i, y_i)$ is the i-th node located at (x_i, y_i) in the image. $e_c = (i,j)$ is the c-th edge from the i-th node to the j-th node. To evaluate the similarity between the two graphs, we first compute affinity matrices $\boldsymbol{K}_v \in \mathbb{R}^{n_1\times n_2}$ and $\boldsymbol{K}_e \in \mathbb{R}^{m_1\times m_2}$ for the similarity of nodes and edges, respectively.

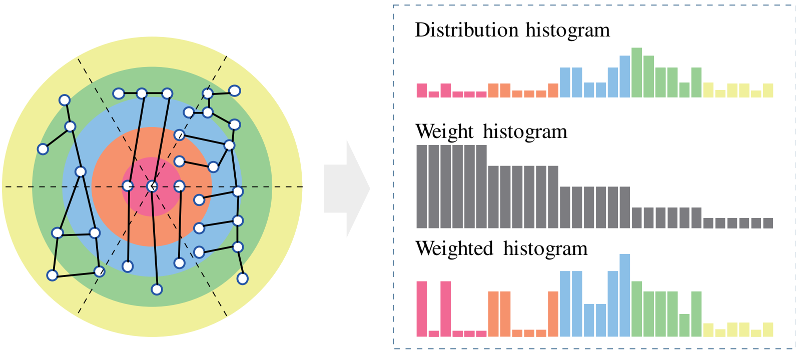

Figure 5 (Color online) Shape context feature distribution of seal character graph.

2.2.1 *Node affinity matrix*

The similarity matrix between the key points is calculated using three features, i.e., spatial location, shape context, and number of edges linked to the key point. The similarity between the i-th node of $\boldsymbol{G}_1$ and the j-th node of $\boldsymbol{G}_2$ is calculated as follows:

$$k_{ij}^{v} = \boldsymbol{W}((v_i^1),(v_j^2)) \cdot \mathrm{e}^{-(\boldsymbol{D}_s(v_i^1,v_j^2)+\boldsymbol{D}_c(v_i^1,v_j^2))}. \tag{5}$$

Here, $\boldsymbol{W}((v_i^1),(v_j^2))$ is a weight measuring the difference in connection numbers between two nodes calculated as follows:

$$\boldsymbol{W}((v_1^i),(v_2^j)) = \begin{cases} 1, & \text{if } |N(v_1^i) - N(v_2^j)| = 0, \\ 0.75, & \text{if } |N(v_1^i) - N(v_2^j)| = 1, \\ 0.5, & \text{if } |N(v_1^i) - N(v_2^j)| = 2, \\ 0.25, & \text{otherwise}, \end{cases} \tag{6}$$

where $N(v_1^i)$ is the number of edges linked to node v_i in $\boldsymbol{G}_1$. $\boldsymbol{D}_s(v_1^i, v_2^j)$ is the Euclidean distance between the i-th node of $\boldsymbol{G}_1$ and the j-th node of $\boldsymbol{G}_2$, which ensures the consistency between the target node and the corresponding matching node in the spatial position. σ_d is the spatial scale factor.

$$\boldsymbol{D}_s(v_i^1, v_j^2) = \|v_i^1 - v_j^2\|_2/\sigma_d. \tag{7}$$

$\boldsymbol{D}_c(v_1^i, v_2^j)$ describes the difference in the context of the nodes as follows:

$$\boldsymbol{D}_c(v_i^1, v_j^2) = \|F_{v_i}^1 - F_{v_j}^2\|_2, \tag{8}$$

where $F_{v_i}^1$ is the context feature of the i-th node in $\boldsymbol{G}_1$. Inspired by the shape context proposed by Belongie et al. [15] and based on observation of the seal characters, we describe the context feature of the node using a weighted histogram. Here, the space around the target node is divided into 6×5 regions, each of which contains six different directions and five different distances (Figure 5).

2.2.2 *Edge affinity matrix*

The pairwise edge affinity matrix $\boldsymbol{K}_e$ is calculated using three edge features, i.e., spatial position, length, and edge orientation. The affinity between the i-th edge in $\boldsymbol{G}_1$ and j-th edge in $\boldsymbol{G}_2$ is calculated as follows:

$$k_{ij}^{e} = \mathrm{e}^{-(\boldsymbol{D}_m(e_i^1,e_j^2)+\boldsymbol{D}_\theta(e_i^1,e_j^2)+\boldsymbol{D}_l(e_i^1,e_j^2)}, \tag{9}$$

where $\boldsymbol{D}_m(e_i^1, e_j^2)$ denotes the Euclidean distance between the midpoints of the two edges, and $\boldsymbol{D}_\theta(e_i^1, e_j^2)$ denotes the difference in orientation between the i-th edge in $\boldsymbol{G}_1$ and j-th edge in $\boldsymbol{G}_2$. $\boldsymbol{D}_l(e_i^1, e_j^2)$ denotes the difference in length between two edges. Specifically, we obtain the following equations:

$$\boldsymbol{D}_m(e_i^1, e_j^2) = \|\mathrm{Mid}(e_i^1) - \mathrm{Mid}(e_j^2)\|_2/\sigma_d,$$

$$\boldsymbol{D}_\theta(e_i^1, e_j^2) = |(\theta(e_i^1) - \theta(e_j^2))|/\sigma_\theta,$$
$$\boldsymbol{D}_l(e_i^1, e_j^2) = |(\mathrm{length}(e_i^1) - \mathrm{length}(e_j^2))|/\sigma_d.$$

Here, $\theta(e_i^1)$ is the direction of the vector from the start point to the terminal point of the i-th edge, and σ_θ is the angle scale factor.

2.3 Graph matching

Figure 6(a) shows an example of matching two seal character graphs. For seal characters, the nodes of their graph model are typically the branch points, turning points, and endpoints of the character strokes and edges are the strokes. Given the two graph models $\boldsymbol{G}_1 = \{\boldsymbol{V}_1; \boldsymbol{E}_1\}$ and $\boldsymbol{G}_2 = \{\boldsymbol{V}_2; \boldsymbol{E}_2\}$, we can compute the global affinity matrix $\boldsymbol{K} \in \mathbb{R}^{n_1 n_2 \times n_1 n_2}$ between the two graph model nodes according to the node feature and the connection feature. According to the global affinity matrix $\boldsymbol{K}$, the graph matching problem is to find a mapping between the two node sets that preserves the relationship between nodes as much as possible by maximizing the global consistency score as follows:

$$\max_{\boldsymbol{X}\in\Pi} \; J_{\mathrm{gm}}(\boldsymbol{X}) = \mathrm{vec}(\boldsymbol{X})^{\mathrm{T}}\mathrm{K}\mathrm{vec}(\boldsymbol{X}). \tag{10}$$

Here, $\boldsymbol{X} \in \Pi$ is constrained as a one-to-one mapping, i.e., Π is the set of partial permutation matrices, as follows:

$$\Pi = \{\boldsymbol{X} | \boldsymbol{X} \in \{0,1\}^{n_1\times n_2}, \boldsymbol{X}\mathbf{1}_{n_2} \leqslant \mathbf{1}_{n_1}, \boldsymbol{X}^{\mathrm{T}}\mathbf{1}_{n_1} = \mathbf{1}_{n_2}\}. \tag{11}$$

The matrix $\boldsymbol{X}$ shown in the left column of Figure 6(d) defines the node correspondence shown in Figure 6(a). The affinity matrix $\boldsymbol{K}$ can be factorized as follows[1)]:

$$\boldsymbol{K} = \mathrm{diag}(\mathrm{vec}(\boldsymbol{K}_v)) + (\boldsymbol{S}_2 \otimes \boldsymbol{S}_1)\mathrm{diag}(\mathrm{vec}(\boldsymbol{K}_e))(\boldsymbol{T}_2 \otimes \boldsymbol{T}_1)^{\mathrm{T}}, \tag{12}$$

where $\boldsymbol{S}$ and $\boldsymbol{T}$ are a source edge matrix and a terminus edge matrix, respectively, that represent the node-edge incidence matrix specifying the topology of the character's graph, as shown in Figure 6(b). For a graph $\boldsymbol{G}$ comprising n nodes and m edges, $\boldsymbol{S} \in \{0,1\}^{n\times m}$ and $\boldsymbol{T} \in \{0,1\}^{n\times m}$ are calculated as follows:

$$s(i,c) = t(j,c) = \begin{cases} 1, & \text{if } e_c = (i,j), \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

Here, $c = 1,2,\ldots,m$ and $i,j = 1,2,\ldots,n$. Figure 6(c) shows an example pair of $\boldsymbol{K}_v$ and $\boldsymbol{K}_e$ for the seal character according to (5) and (9). Combining (12) into (10) yields the following equivalent objective function:

$$J_{\mathrm{gm}}(\boldsymbol{X}) = \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \mathrm{tr}(\boldsymbol{K}_e^{\mathrm{T}}(\boldsymbol{S}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{S}_2 \circ \boldsymbol{T}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{T}_2)), \tag{14}$$

where $\boldsymbol{Y} = \boldsymbol{S}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{S}_2 \circ \boldsymbol{T}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{T}_2$, $\boldsymbol{Y} \in \{0,1\}^{m_1\times m_2}$ can be interpreted as a correspondence matrix for edge pairs, i.e., $y_{c_1c_2} = 1$ if both nodes of the c_1-th edge in $\boldsymbol{G}_1$ are matched to the nodes of c_2-th edge in $\boldsymbol{G}_2$. For example, the right column of Figure 6(d) illustrate the correspondence matrix of edge pairs. By decomposing $\boldsymbol{K}_e$ with $\boldsymbol{K}_e = \boldsymbol{U}\boldsymbol{V}^{\mathrm{T}}$, the above problem can be formulated as follows:

$$\begin{aligned} J_{\mathrm{gm}}(\boldsymbol{X}) &= \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \mathrm{tr}((\boldsymbol{U}\boldsymbol{V}^{\mathrm{T}})^{\mathrm{T}}(\boldsymbol{S}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{S}_2 \circ \boldsymbol{T}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{T}_2)) \\ &= \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \mathrm{tr}\left(\left(\sum_{i=1}^{c} u_i v_i^{\mathrm{T}}\right)^{\mathrm{T}} (\boldsymbol{S}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{S}_2 \circ \boldsymbol{T}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{T}_2)\right) \\ &= \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \sum_{i=1}^{c} \mathrm{tr}(\boldsymbol{A}_i^1\boldsymbol{X}\boldsymbol{A}_i^2\boldsymbol{X}^{\mathrm{T}}), \end{aligned} \tag{15}$$

1) $\boldsymbol{X} \otimes \boldsymbol{Y}$ is the Kronecker product of matrices. $\boldsymbol{X} \circ \boldsymbol{Y}$ is the Hadamard product of matrices. $\mathrm{vec}(\boldsymbol{X})$ denotes the vectorization of matrix $\boldsymbol{X}$. $\mathrm{diag}(x)$ is a diagonal matrix whose diagonal elements are x.

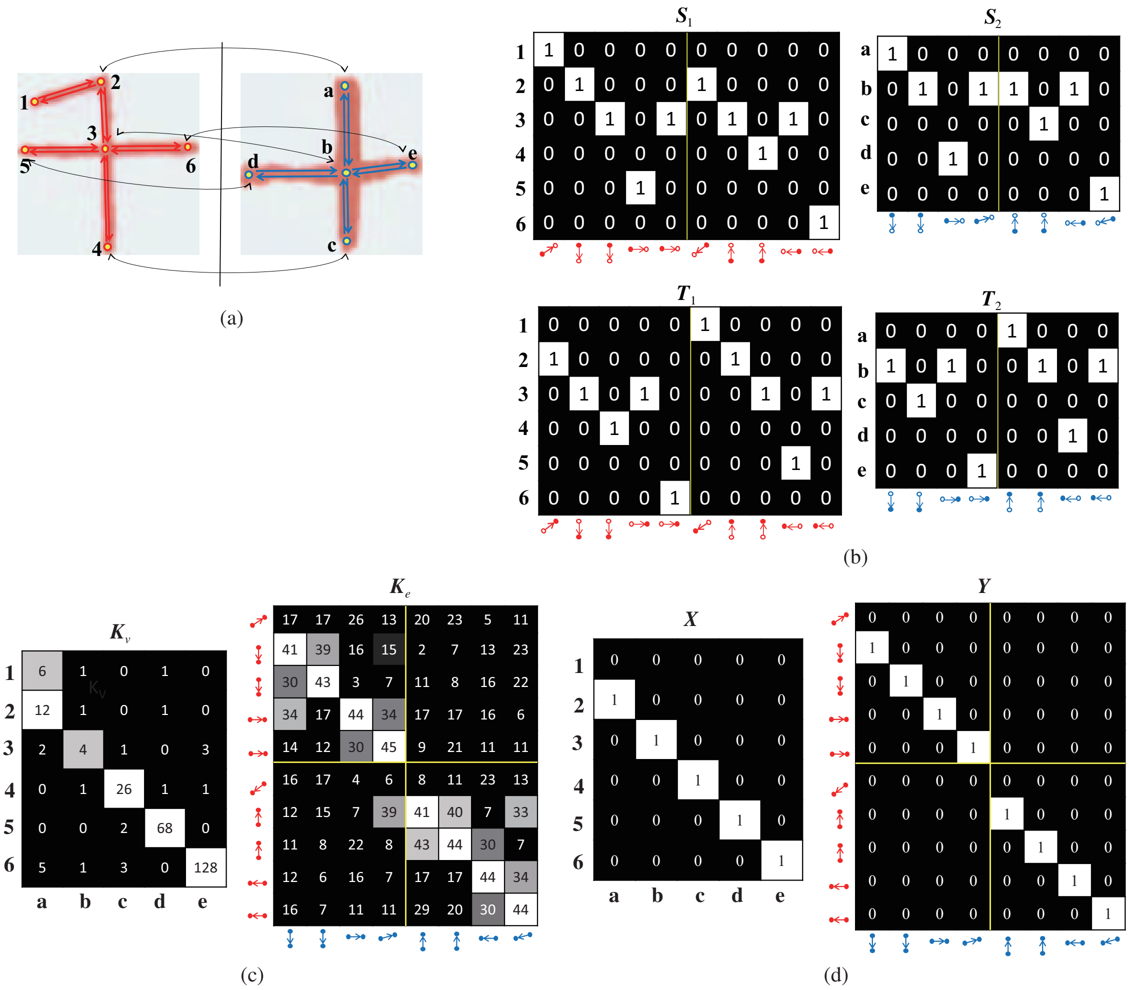

Figure 6 (Color online) An example of seal character matching. (a) Two seal character graphs; (b) two graphs' incidence matrices $\boldsymbol{S}$ and $\boldsymbol{T}$, where non-zero elements in each column of $\boldsymbol{S}$ and $\boldsymbol{T}$ indicate the start and terminal nodes in the corresponding directed edge, respectively; (c) node affinity matrix $\boldsymbol{K}_v$ and edge affinity matrix $\boldsymbol{K}_e$; (d) node correspondence matrix $\boldsymbol{X}$ and edge correspondence matrix $\boldsymbol{Y}$.

where $\boldsymbol{A}_i^1 = \boldsymbol{S}_1\mathrm{diag}(u_i)\boldsymbol{T}_1^{\mathrm{T}} \in \mathbb{R}^{n_1\times n_1}$ and $\boldsymbol{A}_i^2 = \boldsymbol{S}_2\mathrm{diag}(v_i)\boldsymbol{T}_2^{\mathrm{T}} \in \mathbb{R}^{n_2\times n_2}$. As this is an NP-hard problem with permutation matrix $\boldsymbol{X}$, this objective function is relaxed to a convex problem and a concave problem as follows:

$$\begin{aligned} J_{\mathrm{vex}}(\boldsymbol{X}) &= J_{\mathrm{gm}}(\boldsymbol{X}) - \frac{1}{2}J_{\mathrm{con}}(\boldsymbol{X}) = \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) - \frac{1}{2}\sum_{i=1}^{c}\|\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}_i^1 - \boldsymbol{A}_i^2\boldsymbol{X}^{\mathrm{T}}\|_F^2, \\ J_{\mathrm{cav}}(\boldsymbol{X}) &= J_{\mathrm{gm}}(\boldsymbol{X}) + \frac{1}{2}J_{\mathrm{con}}(\boldsymbol{X}) = \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \frac{1}{2}\sum_{i=1}^{c}\|\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}_i^1 - \boldsymbol{A}_i^2\boldsymbol{X}^{\mathrm{T}}\|_F^2, \end{aligned} \tag{16}$$

where

$$J_{\mathrm{con}}(\boldsymbol{X}) = \sum_{i=1}^{c}\mathrm{tr}\big(\boldsymbol{A}_i^{1\,\mathrm{T}}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}_i^1\big) + \mathrm{tr}\big(\boldsymbol{A}_i^2\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{A}_i^{2\,\mathrm{T}}\big), \tag{17}$$

which is constant for any permutation matrix $\boldsymbol{X}$. Here, a new objective is constructed by combining the following relaxations:

$$\max_{\boldsymbol{X}\in\mathcal{D}} \; J_\alpha(\boldsymbol{X}) = (1-\alpha)J_{\mathrm{vex}}(\boldsymbol{X}) + \alpha J_{\mathrm{cav}}(\boldsymbol{X}), \tag{18}$$

$$\mathcal{D} = \{\boldsymbol{X} \in \mathbb{R}^{n_1\times n_2} | \boldsymbol{X}\mathbf{1}_{n_2} \leqslant \mathbf{1}_{n_1}, \boldsymbol{X}^{\mathrm{T}}\mathbf{1}_{n_1} = \mathbf{1}_{n_2}, \boldsymbol{X} \geqslant 0\}.$$

Figure 7 (Color online) Character sample examples.

Here, $\alpha \in [0,1]$ is a tradeoff between the convex and concave relaxations. Optimal $\boldsymbol{X}$ is obtained by gradually increasing α from 0 to 1. During this process, $\boldsymbol{X}$ is updated for different values of α in an iterative manner. For a single α, the modified Frank-Wolfe algorithm [24] is employed to solve $J_\alpha(\boldsymbol{X})$. The optimal $\boldsymbol{X}$ is obtained when $\alpha = 1$. Algorithm 2 summarizes the factorized graph matching algorithm.

Algorithm 2 Factorized graph matching

Input: The node and edge affinity matrices $\boldsymbol{K}_v$, $\boldsymbol{K}_e$; the two input graphs' incidence matrices $\boldsymbol{S}_1$, $\boldsymbol{T}_1$, $\boldsymbol{S}_2$, $\boldsymbol{T}_2$; step length δ; the initial node correspondence matrix $\boldsymbol{X}_0$.

Output: The node correspondence matrix $\boldsymbol{X}$; the matching score between two graphs $\boldsymbol{J}$.

1: Initialize $\boldsymbol{X}$ to be a doubly stochastic matrix;
2: Factorize $\boldsymbol{K}_e = \boldsymbol{U}\boldsymbol{V}^{\mathrm{T}}$;
3: **for** $\alpha = 0 : \delta : 1$ **do**
4:   **if** $\alpha = 0.5$ and $J_{\mathrm{gm}}(\boldsymbol{X}) < J_{\mathrm{gm}}(\boldsymbol{X}_0)$ **then**
5:     Update $\boldsymbol{X} \longleftarrow \boldsymbol{X}_0$;
6:   **end if**
7:   Optimize (18) via MFW to obtain $\boldsymbol{X}^*$;
8:   Update $\boldsymbol{X} \leftarrow \boldsymbol{X}^*$;
9: **end for**
10: Calculate $J_{\mathrm{gm}}(\boldsymbol{X}) = \mathrm{tr}(\boldsymbol{K}_v^{\mathrm{T}}\boldsymbol{X}) + \mathrm{tr}(\boldsymbol{K}_e^{\mathrm{T}}(\boldsymbol{S}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{S}_2 \circ \boldsymbol{T}_1^{\mathrm{T}}\boldsymbol{X}\boldsymbol{T}_2))$;
11: Return $\boldsymbol{X}$, $\boldsymbol{J}$.

---

3 Experiments

3.1 Database and parameter setting

To verify the effectiveness of the proposed Chinese seal character recognition method, we collected 389 seal images containing 1350 character samples from 383 categories. Some seal characters were severely broken and many character categories contain only one sample; thus, we discarded characters with fewer than five samples. Thus, a dataset of 751 seal character samples belonging to 59 categories was obtained. The largest categories comprised 64 samples, and the smallest categories contained only five samples. Figure 7 shows examples of seal character samples from the dataset.

The parameters of the proposed method are listed in Table 1. In the graph model construction process for a Chinese seal character, the path length threshold for the skeleton pruning was set to 10 pixels. In the polygon approximates process, the thresholds for the distance and angle between the turning point and two ends were set to 5 and 145°, respectively. In node feature extraction, the space around the node was partitioned by 60° and the space step was 20 in node feature extraction. In addition, when calculating the affinity matrix between graphs, the scale factor of the spatial distance and orientation were set to 35 and 25°, respectively.

Table 1 Parameter settings

Step	Parameter	Value
Skeleton pruning	Length threshold $l_{\max}$	10
Polygon approximation	Angle threshold $\theta_{\min}$	2.36
	Distance threshold $d_{\min}$	5
Shape feature	Step length	20
	Orientation difference	0.52
Affinity matrix calculating	Spatial scale factor σ_d	35
	Orientation scale factor σ_θ	25

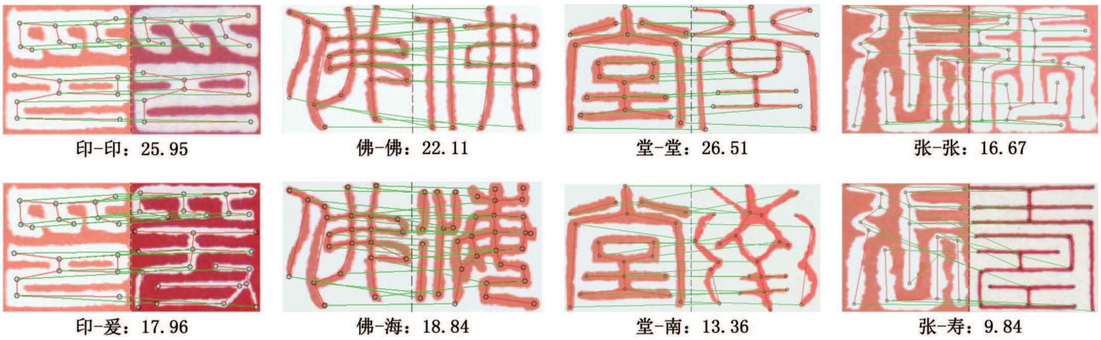

Figure 8 (Color online) Comparison of matching for samples from the same and different categories.

3.2 Experiment results

3.2.1 *Seal character matching*

Figure 8 compares the matching scores for samples from the same and different categories. Here, the top row shows the matching results with the highest score for samples of the same category. As shown, sample pairs from the same category contain similar structural information and most nodes obtain the correct matching. The bottom row shows the matching results with the highest score for samples from different categories. By comparing the matching scores in the top and bottom rows, it can be observed that pairs from different categories obtain low matching scores due to their differing structures. These graph matching results demonstrate that graph matching is effective for the recognition of characters in Chinese seal images.

3.2.2 *Seal character retrieval*

In addition to character recognition, the proposed graph matching method can be used for character retrieval with the similar graph of the target character. Figure 9 shows the top five results of the seal character query. Here, the images to the left of the dashed line are the input characters, and the retrieved reference images with the top five matching scores are shown on the right of the dashed line. The characters in the input images are “之 (zhi)”, “堂 (tang)”, “寿 (shou)”, “张 (zhang)”, “印 (yin)”, “爰 (yuan)”, “书 (shu)”, and “海 (hai)” from top to bottom and left to right. The number under each retrieved result is the matching score (with the input image). As can been seen, the proposed method yields accurate results for most input images. Note that the matching score varies for different categories relative to the number of nodes in the graph models and geometric similarity. The number under the input image is the overall recognition rate of the corresponding category. To evaluate the seal character recognition performance on the whole dataset, each sample was taken as an input image and the others as references. The results of an objective evaluation of retrieval performance are shown in Table 2 in terms of the top 1, 3, and 5 accuracies. As can be seen, the proposed method demonstrates promising accuracy for Chinese seal character retrieval.

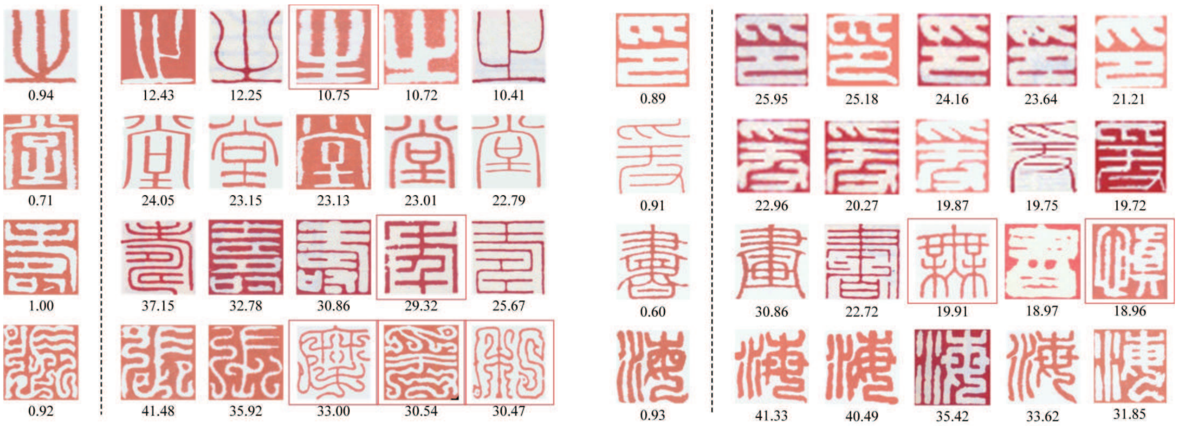

Figure 9 (Color online) The examples of seal character retrieval. The input images are on the left side of the dashed line. The character recognition rate is at the bottom of each input image. The retrieval results are on the right side of the dashed line sorted in descending order of matching scores.

Table 2 Recognition accuracy of top 1, 3 and 5 with the proposed method

	Top 1	Top 3	Top 5
Accuracy (%)	83.42	88.52	91.33

3.2.3 *Comparison with other methods*

For each category, 50% of the samples was selected as training samples or references, whereas the remaining 50% was used as test characters. The proposed method was compared with several popular classification methods, i.e., support vector machine (SVM) [25], convolutional neural network (CNN) [26], matching pursuit-based subspace clustering (MPSC) [27], and sparse representation classification (SRC) [28] methods. The recognition results for all methods are shown in Table 3. The numbers in Table 3 indicate the number of test, training, and correctly recognized samples. The overall accuracies of the different methods are given in the last column. The performance of the SVM was poor due to limited and imbalanced training samples (Table 3). The MPSC and SRC methods achieved relatively better classification accuracy because they do not have strict requirements relative to the number and distribution of samples. Note that the CNN architecture is the same as that reported in the literature [26]. The performance of Yann's CNN is better than the above three methods but worse than that of the proposed method with limited and imbalanced training samples. The CNN performance may be improved if innovative activation [29] or an appropriate network structure is used to handle insufficient and imbalanced training samples. Compared to the above four methods, the proposed method achieved higher recognition accuracy, which demonstrates the advantage of the proposed method on a dataset with a small number of training samples. In addition, these results demonstrate that the proposed method is robust to an imbalanced sample distribution. As can be seen in Table 3, the proposed method achieved the best recognition results for nearly all the categories. The geometrical structure of the seal character used in the proposed method is very robust against local variations. Since the geometrical structure is invariant to the appearance of the character, e.g., stroke width, no additional extra training samples are required unless the geometrical structure varies significantly.

3.3 Effect of parameters

The experimental results demonstrate that the proposed algorithm works very well with the parameters given in Table 1. Here, we discuss experiments conducted to evaluate the effect of different parameter values. Only one parameter was varied in each experiment (all other variables were used as given in Table 1).

The proposed method uses the spatial distance scale factor σ_d and angle scale factor σ_θ to control the weight of the spatial position feature and the connection angle feature when calculating the correspon-

Table 3 Recognition results of different methods

Category	不	个	之	书	二	人	佛	刘	北	千	午	南	印	启	堂
Test/train	4/3	3/3	17/17	5/5	4/4	8/7	3/2	7/7	3/2	25/24	3/2	3/2	19/19	6/5	7/7
SVM [25]	0	0	14	0	3	1	0	1	0	12	2	0	4	1	0
CNN [26]	2	0	10	0	4	**5**	0	6	0	17	1	1	16	2	4
MPSC [27]	2	0	11	1	4	4	1	5	0	14	1	1	14	0	0
SRC [28]	2	0	11	1	4	4	1	5	0	15	1	1	14	0	0
GMCSCR	**4**	0	**16**	**4**	4	4	**3**	**7**	0	**21**	**3**	**2**	**18**	**6**	**5**

Category	墨	士	大	天	好	季	宁	家	寿	居	山	己	巳	年	庐
Test/train	3/2	5/5	32/32	3/3	3/2	3/2	7/6	3/3	5/4	6/5	3/2	3/2	3/2	3/3	3/2
SVM [25]	0	2	9	0	1	0	0	0	0	0	1	1	0	0	0
CNN [26]	0	3	26	0	0	0	0	0	3	3	2	1	0	0	0
MPSC [27]	1	2	30	0	0	0	0	0	0	**5**	**3**	1	0	0	0
SRC [28]	1	3	30	0	0	0	0	0	0	**5**	**3**	1	0	0	1
GMCSCR	**2**	**4**	**31**	**3**	**2**	**1**	**6**	**1**	**5**	4	2	**2**	0	**1**	**3**

Category	张	心	戊	成	我	摩	斧	海	爱	王	生	画	真	石	私
Test/train	26/25	3/3	3/2	5/5	4/3	3/3	8/7	21/21	22/22	6/5	3/3	6/5	3/2	3/3	5/5
SVM [25]	3	0	0	0	**2**	0	0	2	7	5	0	0	0	0	0
CNN [26]	19	0	0	1	1	0	3	19	15	1	0	2	1	0	3
MPSC [27]	20	0	0	1	1	1	4	12	14	3	0	1	0	0	**4**
SRC [28]	20	0	0	1	1	1	4	12	14	3	0	1	0	0	**4**
GMCSCR	**24**	**1**	**1**	**4**	1	**3**	**5**	**21**	**20**	**6**	**3**	2	**3**	**2**	3

Category	簃	粟	翁	老	腐	花	藏	蜀	西	贤	进	鉢	长	风	Overall Accuracy (%)
Test/train	3/2	13/13	7/6	7/7	3/2	3/2	3/2	3/3	4/3	5/5	3/2	3/3	7/6	5/5	
SVM [25]	0	1	0	0	0	0	0	0	0	0	0	0	0	0	18.37
CNN [26]	0	9	3	1	0	0	1	1	1	2	0	1	2	2	49.23
MPSC [27]	0	9	1	0	1	0	1	0	1	2	1	1	1	1	46.17
SRC [28]	0	9	1	0	1	0	1	1	1	3	0	2	1	1	47.45
GMCSCR	**3**	**12**	**7**	**7**	**2**	0	**3**	1	**4**	**5**	1	2	**5**	**3**	**83.42**

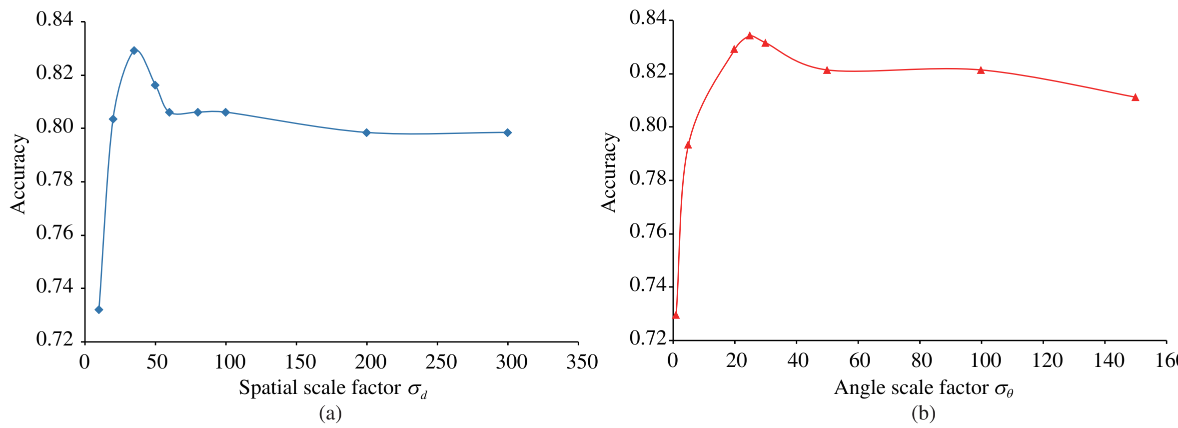

Figure 10 (Color online) Effect of different parameter values on performance. (a) Spatial scale factor σ_d; (b) angle scale factor σ_θ.

dence matrix. A greater scale factor value leads to a smaller weight for the corresponding feature.

Figure 10 shows the recognition accuracy of the proposed method for different σ_d and σ_θ values. As shown in Figure 10(a), the proposed method achieved the highest recognition accuracy when σ_d was 35. When σ_d was too small, i.e., $\sigma_d < 35$, the spatial position feature may be overly emphasized in the calculation of the correspondence matrix, which may reduce the robustness of the proposed method against variances in character structure and result in reduced recognition accuracy. When σ_d was increased gradually from 35, the weight of the spatial position feature gradually decreased until its influence on the graph matching finally disappeared. The result with very large σ_d was an approximation of the result without spatial position feature. By comparing the results with $\sigma_d = 35$ and $\sigma_d = 300$, we conclude that the spatial position feature is a valid and useful feature that may improve performance with a proper weight. Therefore, σ_d was set to 35 in the proposed method. Figure 10(b) shows that the angle scale factor works in a similar manner as spatial scale factor σ_d. Thus, σ_θ was set to 25 for optimal performance.

4 Conclusion

This study has proposed a Chinese seal character recognition method based on graph matching. In the proposed method, the structure of the character is extracted with skeletonization and polygonal approximation. Then, end, branch, and turning points are detected and denoted as the nodes of the graph, and strokes are modeled with the edges of the graph. The affinity matrices of the nodes and edges are then calculated using spatial, contextual, and geometrical features, and with the two affinity matrices, the correspondence matrix between the two graphs is estimated using a factorized graph matching algorithm. In addition, a reference character image with the highest matching score determines the category of the input image. The proposed method achieves higher recognition accuracy than several existing methods on a seal character dataset. The experimental results demonstrate the superiority of the proposed method over several classifiers when using a small training dataset.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61520106001, 61801178), Natural Science Foundation of Hunan Province (Grant No. 2018JJ3071), and by Hunan Key Laboratory of Visual Perception and Artificial Intelligence.

References

1 Roy P P, Pal U, Lladós J. Document seal detection using GHT and character proximity graphs. Pattern Recogn, 2011, 44: 1282–1295
2 Ren C, Liu D, Chen Y B. A new method on the segmentation and recognition of Chinese characters for automatic Chinese seal imprint retrieval. In: Proceedings of International Conference on Document Analysis and Recognition, 2011. 972–976
3 Roy P P, Pal U, Lladós J. Seal detection and recognition: an approach for document indexing. In: Proceedings of International Conference on Document Analysis and Recognition, 2009. 101–105
4 Yin F, Wang Q F, Zhang X Y, et al. Chinese handwriting recognition competition. In: Proceedings of International Conference on Document Analysis and Recognition, 2013. 1464–1469
5 Wang C H, Xiao B H, Dai R W. Parallel compact integration in handwritten Chinese character recognition. Sci China Ser F-Inf Sci, 2004, 47: 89–96
6 Guo J, Wang C H, Roman-Rangel E, et al. Building hierarchical representations for Oracle character and sketch recognition. IEEE Trans Image Process, 2016, 25: 104–118
7 Sebastian T B, Klein P N, Kimia B B. Recognition of shapes by editing their shock graphs. In: Proceedings of IEEE International Conference on Computer Vision, 2011. 755–762
8 Bai X, Latecki L J. Path similarity skeleton graph matching. IEEE Trans Pattern Anal Mach Intell, 2008, 30: 1282–1292
9 Zhang H, Mu Y, You Y H, et al. Multi-scale sparse feature point correspondence by graph cuts. Sci China Inf Sci, 2010, 53: 1224–1232
10 Zhang L M, Yang Y, Wang M, et al. Detecting densely distributed graph patterns for fine-grained image categorization. IEEE Trans Image Process, 2016, 25: 553–565
11 Mian A S, Bennamoun M, Owens R. Three-dimensional model-based object recognition and segmentation in cluttered scenes. IEEE Trans Pattern Anal Mach Intell, 2006, 28: 1584–1601
12 Jiang H, Yu S X, Martin D R. Linear scale and rotation invariant matching. IEEE Trans Pattern Anal Mach Intell, 2011, 33: 1339–1355
13 Zhao R, Martinez A M. Labeled graph kernel for behavior analysis. IEEE Trans Pattern Anal Mach Intell, 2016, 38: 1640–1650
14 Aksoy E E, Abramov A, Worgotter F, et al. Categorizing object-action relations from semantic scene graphs. In: Proceedings of IEEE International Conference on Robotics and Automation, 2010. 398–405
15 Belongie S, Malik J. Matching with shape contexts. In: Proceedings Workshop on Content-based Access of Image and Video Libraries, 2000

16 Liu C L, Kim I J, Kim J H. Model-based stroke extraction and matching for handwritten Chinese character recognition. Pattern Recogn, 2001, 34: 2339–2352
17 Fischler M A, Bolles R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Read Comput Vision, 1987, 24: 726–740
18 Schenker P S. Method for registration of 3-D shapes. IEEE Trans Pattern Anal Mach Intell, 2002, 14: 239–256
19 Cho M, Lee J, Lee K M. Reweighted random walks for graph matching. In: Proceedings European Conference on Computer Vision, 2010. 492–505
20 Mateus D, Horaud R, Knossow D, et al. Articulated shape matching using Laplacian eigenfunctions and unsupervised point registration. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2008
21 Zhou F, De la Torre F. Factorized graph matching. IEEE Trans Pattern Anal Mach Intell, 2016, 38: 1774–1789
22 Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern, 1979, 9: 62–66
23 Wang Y, Zhong B J. A scale-space technique for polygonal approximation of planar curves. In: Proceedings of IEEE International Conference on Image Processing, 2013. 517–520
24 Fukushima M. A modified Frank-Wolfe algorithm for solving the traffic assignment problem. Transpation Res Part B-Meth, 1984, 18: 169–177
25 Chang C C, Lin C J. LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol, 2011, 2: 1–27
26 Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proc IEEE, 1998, 86: 2278–2324
27 You C, Robinson D P, Vidal R. Scalable sparse subspace clustering by orthogonal matching pursuit. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016. 3918–3927
28 Qu X W, Wang W Q, Lu K. In-air handwritten Chinese character recognition using discriminative projection based on locality-sensitive sparse representation. In: Proceedings of International Conference on Pattern Recognition, 2017. 1137–1140
29 Cao J L, Pang Y W, Li X L, et al. Randomly translational activation inspired by the input distributions of ReLU. Neurocomputing, 2018, 275: 859–868