Vis Comput
DOI 10.1007/s00371-016-1303-3

ORIGINAL ARTICLE

Fast algorithm for 2D fragment assembly based on partial EMD

Meng Zhang[1] · Shuangmin Chen[1] · Zhenyu Shu[2] · Shi-Qing Xin[1] · Jieyu Zhao[1] · Guang Jin[1] · Rong Zhang[1] · Jürgen Beyerer[3]

Abstract 2D Fragment assembly is an important research topic in computer vision and pattern recognition, and has a wide range of applications such as relic restoration and remote sensing image processing. The key to this problem lies in utilizing contour features or visual cues to find the optimal partial matching. Considering that previous algorithms are weak in predicting the best matching configuration of two neighboring fragments, we suggest using the earth mover's distance, based on length/property correspondence, to measure the similarity, which potentially matches a point on the first contour to a desirable destination point on the second contour. We further propose a greedy algorithm for 2D fragment assembly by repeatedly assembling two neighboring fragments into a composite one. Experimental results on map-piece assembly and relic restoration show that our algorithm runs fast, is insensitive to noise, and provides a novel solution to the fragment assembly problem.

Keywords Fragment assembly · Partial EMD · Contour features · Lebesgue measure

✉ Shuangmin Chen
chenshuangmin@nbu.edu.cn

[1] Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, People's Republic of China

[2] School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, People's Republic of China

[3] Fraunhofer-Institute of Optronics, System Technologies and Image Exploitation IOSB, Karlsruhe, Germany

1 Introduction

Fragment assembly [21,33,36] based on contour features or chromatic cues is an important research topic in computer vision and pattern recognition [10,25,40,47]. It is of practical use in relic restoration, puzzle assembly [16,19] and remote sensing image processing [29]. In both 2D and 3D cases, most of conventional algorithms [5,6,12,13,24,34,42] require the following steps: (1) preprocessing, (2) feature extraction, (3) computing similarity matrix and (4) assembling fragments. Generally, two kinds of information on the contours, including geometric features (curvatures, torsions, normals) and chromatic cues (texture), are helpful in inferring the assembly configuration.

Without doubt, the underlying measure defining the similarity between neighboring fragments is central to obtain a meaningful assembly configuration. It is observed that when the common boundary is maximized w.r.t. a certain alignment, its length provides a natural measure of similarity. Mathematically speaking, the similarity between neighboring fragments can be defined to the maximum possible Lebesgue measure of the common contour. Suppose that l_1 is the subsegment of the first contour defining the common contour and l_2 is the counterpart of l_1 given by the second contour. Ideally, l_1 and l_2 have no difference except going in the opposite direction. Considering that the input fragments often have noisy boundaries, it is reasonable to take l_1 and l_2 as the common boundary if some specified distance between l_1 and l_2 is less than a given tolerance.

To our best knowledge, most of the known algorithms compute pointwise distances to measure the difference between two curved segments. For example, the Hausdorff distance was employed in [3,4] to find the two points that maximize the deviation, while the Fréchet distance [2,9, 14,28] was used to find the maximum distance between

Published online: 23 August 2016

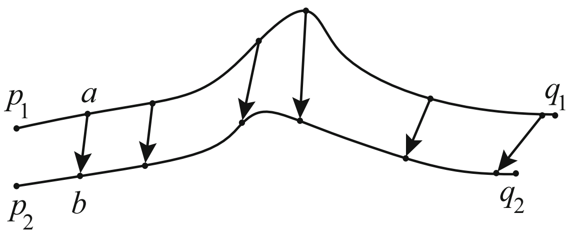


Fig. 1 The earth mover's distance based on length/property correspondence

corresponding points up to some curve re-parametrization. However, these approaches do not take into account the property (e.g., curvatures or visual cues) correspondence and are thus weak in predicting a good matching configuration between two fragments.

As Fig. 1 shows, $\wideparen{p_1q_1}$ and $\wideparen{p_2q_2}$ are two given curves. To establish a desirable correspondence between $\wideparen{p_1q_1}$ and $\wideparen{p_2q_2}$, we need to consider at least two aspects in the following:

- *Length correspondence* If $a \in \wideparen{p_1q_1}$ is mapped to $b \in \wideparen{p_2q_2}$, then it would be better if $\wideparen{p_1a}$ and $\wideparen{p_2b}$ are as equal as possible in length.
- *Property correspondence* If $a \in \wideparen{p_1q_1}$ is mapped to $b \in \wideparen{p_2q_2}$, then it would be better if a and b have similar properties.

In this paper, we suggest using the well known earth mover's distance (EMD) [1,20,37–39,44] to measure the overall difference between two curved segments. Taking Fig. 1 as an example, we define the cost for moving a unit amount of sand from a to b as follows:

$$\begin{aligned}\mathrm{Cost}(a,b) &= \lambda_1 \mid \|\wideparen{p_1a}\| - \|\wideparen{p_2b}\| \mid \\ &\quad + \lambda_2 \mid \mathrm{Property}(a) - \mathrm{Property}(b) \mid, \end{aligned} \tag{1}$$

where the first term measures the length correspondence, the second term measures the property correspondence, and λ_1 and λ_2 are two parameters that make a balance between the two terms. With the help of these preparations, the overall difference between the two curved segments can be defined to the lowest moving cost given by the optimal transport plan. By finding the the maximum possible Lebesgue measure of the common contours whose EMD is less than a given threshold, we are able to find the best configuration between two neighboring fragments. Our contributions are threefold:

1. We suggest using the earth mover's distance, based on length/property correspondence, to measure the similarity between two neighboring fragments. Technically, it can be formulated as a linear programming (LP) problem.
2. Considering that the coherence of matching configurations, we suggest a set of techniques to speed up the computation.
3. We illustrate its effectiveness by demonstrating its uses in puzzle assembly and relic restoration.

2 Related work

At least two topics, i.e., curve matching and fragment assembly, are closely related to our research work in this paper. We summarize the related work in the following.

Curve matching This is an important computational task in research domains such as reconstruction of archeological fragments, forensic investigation, measurement of melodic similarity, and model-based object recognition [29]. As Buchin et al. [9] pointed out, curve matching is actually to maximize the total length of subcurves that are close to each other, where closeness is measured by some 2D distance measure, e.g., Fréchet distance [2,9,14,28] and Hausdorff distance [3,4].

Rather than matching two curves globally, we are more interested in the partial curve matching problem [26], i.e., given a threshold δ, we wish to find the best matching configuration such that the largest possible fractions of two input curves are matched within distance δ. For such a purpose, James [23] suggested capturing the locations of important features which may represent local behavior and then equating the "moments" of a given set of curves. Huang and Cohen [22] used a new class of weighted B-spline curve moments to handle the affine transformation and/or occlusion between the curves. Porrill and Pollard [35] discussed epipolar geometry-based curve matching in camera calibration. Cui et al. [11] proposed a curve matching framework based on a scale-invariant signature.

It is worth noting that in this paper that we consider the length correspondence and property correspondence at the same time when establishing the transport plan between two input curve segments, and then use EMD to evaluate the similarity between them. Our approach is invariant to rigid transformations, insensitive to noise and able to support multiple property functions at the same time.

Fragment assembly The assembly problem discussed in this paper is quite similar to, but different from the conventional puzzle assembly problem [31,41] where the pieces are often symmetric and regular. We assume that the input fragments are of irregular shapes and thus the computation has to depend on shape cues. To assemble them together, it is a common practice to use the greedy assembly strategy [8,17,18,30,32,46,48]. For example, da Gama Leito and Stolfi [17] suggested looking for initial matchings at the coarsest possible scale and then repeatedly selecting the most promising pairs to re-match them at the next finer scale. In this way, they asymptotically reduced the matching cost until the assembly was complete.

In this paper, we also use a greedy algorithm to accomplish the assembly task—fragments that have larger similarity are given higher priority during the assembly process. The key speedup technique different from the conventional greedy scheme is: after fragment #A and fragment #B are combined into a new fragment #C, the properties of #C are updated at only a little cost using those of #A and #B.

3 Partial EMD-based similarity measure

We let

$$p_1^{(i)}, p_2^{(i)}, \ldots, p_{m_i}^{(i)}$$

constitute a closed polygonal curve bounding the i-th fragment, and

$$w_1^{(i)}, w_2^{(i)}, \ldots, w_{m_i}^{(i)}$$

be the corresponding weighting scheme. Typically, $w_j^{(i)}$ is the length of the influential interval by $p_j^{(i)}$. We use the vector

$$Q_1^{(i)}, Q_2^{(i)}, \ldots, Q_{m_i}^{(i)},$$

to denote the geometric or chromatic properties of $p_j^{(i)}$, $j = 1, 2, \ldots, m_i$.

Given two fragments shown in Fig. 2, we discuss the definition of EMD-based similarity between them. For convenience of formulation, we assume that the orientations of the two contours are different from each other. Now, we shall study whether the curved segment from $p_{j_1}^{(1)}$ to $p_{j_1+k_1}^{(1)}$ can be matched with the curved segment from $p_{j_2}^{(2)}$ to $p_{j_2+k_2}^{(2)}$.

First, to guarantee that the information (length and properties) of $\widetilde{p_{j_1}^{(1)} p_{j_1+k_1}^{(1)}}$ can be completely transported to $\widetilde{p_{j_2}^{(2)} p_{j_2+k_2}^{(2)}}$, we should normalize the weights such that the total weights for $\widetilde{p_{j_1}^{(1)} p_{j_1+k_1}^{(1)}}$ and $\widetilde{p_{j_2}^{(2)} p_{j_2+k_2}^{(2)}}$ are exactly 1, even if $\widetilde{p_{j_1}^{(1)} p_{j_1+k_1}^{(1)}}$ and $\widetilde{p_{j_2}^{(2)} p_{j_2+k_2}^{(2)}}$ are not equal in length. In the following, we assume that

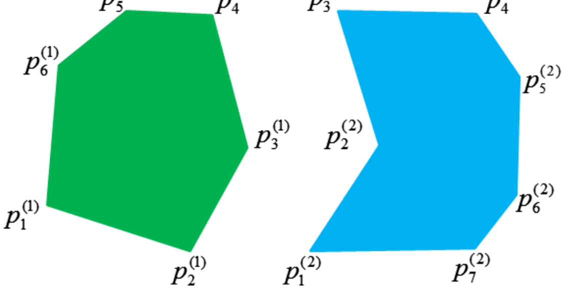


Fig. 2 EMD-based similarity

$$w_{j_1}^{(1)} + w_{j_1+1}^{(1)} + \cdots + w_{j_1+k_1}^{(1)} = 1$$

and

$$w_{j_2}^{(2)} + w_{j_2+1}^{(2)} + \cdots + w_{j_2+k_2}^{(2)} = 1.$$

Now, we assume the information of a mass $\delta_{j^{(1)}j^{(2)}}$ be moved from the vertex $p_{j^{(1)}}^{(1)}$ to the vertex $p_{j^{(2)}}^{(2)}$, whose cost is

$$\begin{aligned}\mathrm{Cost}_{j^{(1)}j^{(2)}} = \delta_{j^{(1)}j^{(2)}} * (&\lambda_1 * |l_{j^{(1)}} - l_{j^{(2)}}| \\ &+ \lambda_2 * |Q_{j^{(1)}} - Q_{j^{(2)}}|),\end{aligned}$$

where $l_{j^{(1)}}$ and $l_{j^{(2)}}$ are, respectively, the lengths from the starting points to $p_{j^{(1)}}^{(1)}$ and $p_{j^{(2)}}^{(2)}$. Our goal is to find some transport plan so as to minimize the following cost function:

$$\mathrm{Cost}(j_1, k_1, j_2, k_2) = \min_{\delta_{j^{(1)}j^{(2)}}} \sum_{j^{(1)}=j_1}^{j_1+k_1} \sum_{j^{(2)}=j_2}^{j_2+k_2} \mathrm{Cost}_{j^{(1)}j^{(2)}},$$

subject to the following constraints:

$$\sum_{j^{(2)}} \delta_{j^{(1)}j^{(2)}} = w_{j^{(1)}}^{(1)},$$

$$\sum_{j^{(1)}} \delta_{j^{(1)}j^{(2)}} = w_{j^{(2)}}^{(2)}$$

and

$$\delta_{j^{(1)}j^{(2)}} \geq 0,$$

where $j^{(1)} = j_1, j_1 + 1, \ldots, j_1 + k_1$, $j^{(2)} = j_2, j_2 + 1, \ldots, j_2 + k_2$ and $\delta_{j^{(1)}j^{(2)}}$ define the transport plan.

Recall that we have normalized the total lengths of the respective curved segments to be one in the very beginning. Therefore, we take

$$\begin{aligned}\mathrm{Cost}^*(j_1, k_1, j_2, k_2) &= \mathrm{Cost}(j_1, k_1, j_2, k_2) \\ &\quad \times \max\left(\left\|\widetilde{p_{j_1}^{(1)} p_{j_1+k_1}^{(1)}}\right\|, \left\|\widetilde{p_{j_2}^{(2)} p_{j_2+k_2}^{(2)}}\right\|\right)\end{aligned}$$

as the real EMD between $\widetilde{p_{j_1}^{(1)} p_{j_1+k_1}^{(1)}}$ and $\widetilde{p_{j_2}^{(2)} p_{j_2+k_2}^{(2)}}$. Finally, the overall similarity between the two input fragments can be defined to be

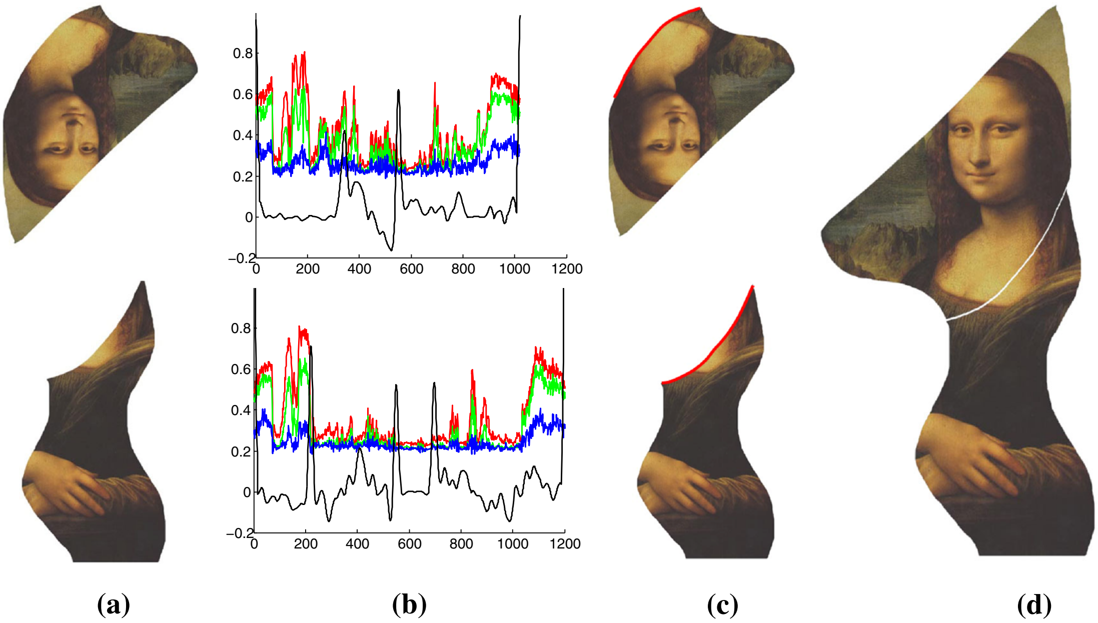


Fig. 3 Algorithmic pipeline: we first extract the properties (**b**) from the input fragments (**a**) and then find the optimal partial matching (**c**) such that they can be assembled (**d**). Note that in **b**, the *colored curves*, respectively, show the change of the three color components (R, G, B) along the boundary, while the *black curve* is the curvature plot

$$L = \max_{j_1,k_1,j_2,k_2} \{\min \left(\left\| \widetilde{p^{(1)}_{j_1} p^{(1)}_{j_1+k_1}} \right\|, \left\| \widetilde{p^{(2)}_{j_2} p^{(2)}_{j_2+k_2}} \right\| \right) \\ \mid \text{Cost}^*(j_1, k_1, j_2, j_{k_2}) \leq \epsilon\}, \qquad (2)$$

where ϵ is the user-specified tolerance to define the maximum allowable EMD when matching two curved segments. That is to say, if the EMD is less than ϵ, we can say that the two input curved segments are well matched. ϵ is experimentally set to 0.3 % of the total length of the input closed curves.

4 Fragment assembly

In this section, we first show the whole algorithmic pipeline of the fragment assembly in Fig. 3, followed by the key steps that are detailed in Sects. 4.1–4.4. The first step is to preprocess the input fragments (see Fig. 3a), including boundary smoothing and (geometric and chromatic) feature extraction (see Fig. 3b). Next, we propose to find the optimal matching (see Fig. 3c) based on the similarity measure discussed in Sect. 3. After that, we further infer the rigid transformation matrix so that the input fragments can be well aligned (see Fig. 3d). Finally in Sect. 4.4, we suggest a greedy assembly scheme to assemble multiple input fragments, i.e., maintaining a priority queue throughout the algorithm so that the top priority fragment pair can be assembled during each step.

4.1 Preprocessing

During the preprocessing step, one key task is to transform the boundary into vectorized representation, typically a closed polygonal curve without noise. First, we use a quadtree-based method proposed by Wang et al. [43] to reconstruct a 2D curve from a set of unorganized points, possibly with a high level of noise and finally produce a desirable contour while preserving geometric features. Next, we further simplify the contours and remove those vertices without which the Hausdorff distance between the original polyline and the simplified version is still within a given tolerance. Here, we take the tolerance to be 0.1 % of the curve length. The simplification operation [15] is for the purpose of accelerating computation. After that, we use the multiscale curvature method proposed by Liu et al. [27] to estimate the curvatures of a polygonal curve. The significant advantage of this method is the ability to yield stable estimation even under severe noise, since the authors considered statistics of the extreme points of the height functions computed over all directions. After that, we can easily obtain the curvature plot with regard to the arc-length parameter; see Fig. 4. The chromatic cues along the boundary, typically consisting of the R, G, B components, can also be encoded by plots. In this way, we finish the property extraction task that is central to the similarity estimation between fragments.

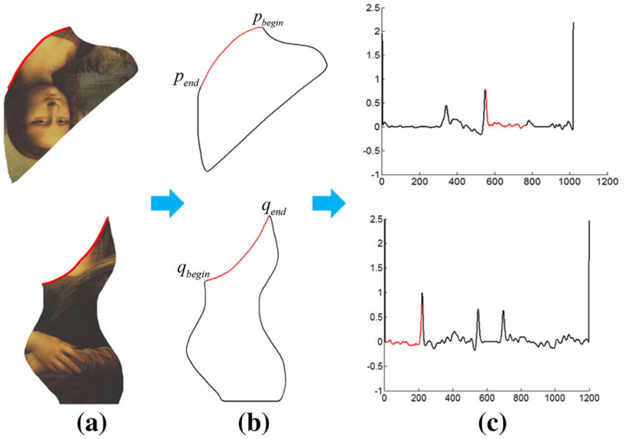


Fig. 4 Curvature extraction. The segments marked in *red* are the common boundary computed by EMD. **a** Fragments. **b** Contours. **c** Curvature plots

4.2 Similarity evaluation

Once we obtain the curvature plots of the fragments, we come to estimate the similarity between two fragments based on the EMD formulation [see Eq. (2)], where the underlying metric is defined as Eq. (1).

The key is to find a proper quadruple (j_1, k_1, j_2, k_2) corresponding to the longest candidate common boundary $\overset{\frown}{p^{(1)}_{j_1} p^{(1)}_{j_1+k_1}}$ (resp. $\overset{\frown}{p^{(2)}_{j_2} p^{(2)}_{j_2+k_2}}$) with its EMD being equal to or less than the given tolerance. A naïve method is to find the best matching among all the possible quadruples, which is very time consuming. Therefore, we propose a pair of techniques as follows:

- In Eq. (1), increasing λ_2 tends to align features between the two given fragments. In this way, it enables us to capture the quite limited number of "moments" that the fragments align. After that, it is sufficient to consider a few possibilities of matching configurations and then enlarge or shrink the common boundary while keeping the feature points aligned.
- The EMD cost changes continuously with regard to the four indices of the quadruple. When we intend to enlarge the common boundary, it is better to change the index component of the quadruple that increases the EMD cost most inconspicuously; when we shrink the common boundary, it is better to change the index component of the quadruple that reduces the EMD cost most significantly.

In implementation, we set λ_1 to 0.1 and λ_2 to 0.9 when capturing alignment moments, and set λ_1 to 0.5 and λ_2 to 0.5 when computing the real EMD cost.

4.3 Alignment matrix

After we find the longest common boundary between the two fragments, we also need to compute an alignment matrix such that they are well placed along the same boundary; see the red curves in Fig. 3c. Now, we compute the alignment by solving a linear system.

Suppose that (x_i, y_i) and (x_i', y_i') are two groups of points that need to be aligned. It is reasonable to assume that a rigid transform is able to match them well. Therefore, we have

$$\begin{cases} x_i^{'} = ax_i + cy_i + t_1, \\ y_i^{'} = bx_i + dy_i + t_2, \end{cases} \tag{3}$$

for $i = 1, 2, \ldots, k$. Generally, the linear equation system (3) is overconstrained whose least square solution actually defines a matrix that is able to align the two fragments; see Fig. 3d for illustration.

4.4 Assembly scheme

To assemble a group of fragments, the first step is to compute a similarity matrix that encodes the similarity between any pair of two fragments. We then use a greedy scheme to assemble all the fragments. Figure 5 illustrates how three pieces of the Mona Lisa picture are assembled step by step. During the first step, Fragment #1 and Fragment #3 are matched since they have a long common boundary according to the 3 × 3 similarity matrix. Therefore, they are assembled into Fragment #1, 3. Let l_{12} be the length of the common boundary between Fragment #1 and Fragment #2, and l_{32} be the common boundary between Fragment #3 and Fragment #2. Then the length of the common boundary between Fragment #1, 3 and Fragment #2 is updated to $l_{12} + l_{32}$. Note that at this moment, the similarity matrix is 2×2. Repeat such a process until all fragments are assembled together; see Fig. 5.

5 Experimental results

We implemented and experimented with our algorithm on a computer with a 64-bit version of Win8 system, a 2.4 GHz Intel(R) Core(TM) i7-5500U CPU and 6 Gb memory. The coding language was C++.

5.1 Performance

We use the fragments in Fig. 4 to test the performance with regard to the number of vertices on the boundary. For such a purpose, we discretize both the boundaries into 1K, 2K, . . ., 35K vertices and run our algorithm. The timing plots are available in Fig 6, where we use $T1$ to denote the time cost (in seconds) for finding the longest common boundary

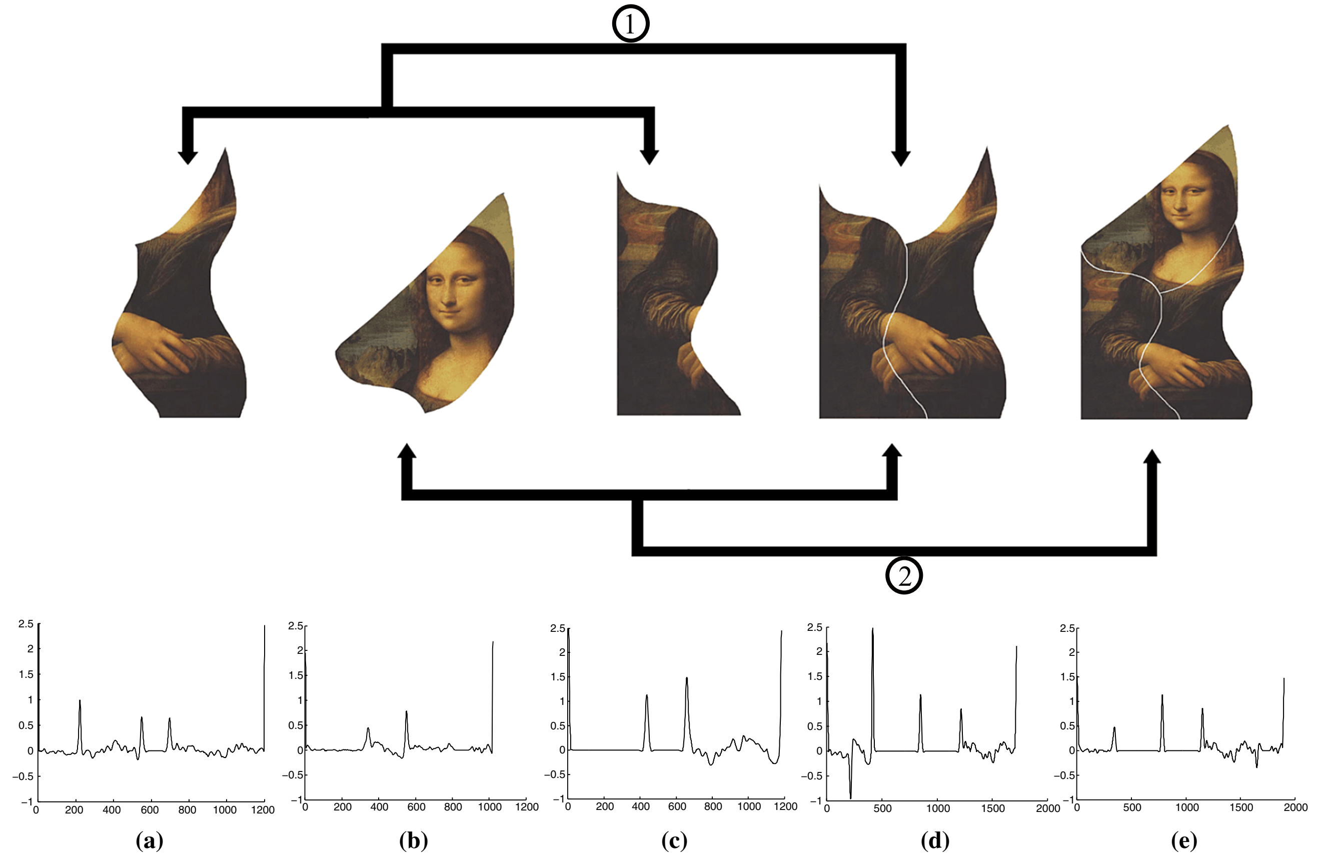


Fig. 5 Greedy algorithmic scheme: the input fragments are assembled together according to the priority of similarity. During the first step, Fragment #1 and Fragment #3 are matched and the properties of the newly generated piece Fragment #1, 3 are updated immediately. After that, Fragment #1, 3 and Fragment #2 are then assembled, resulting in another new piece Fragment #1, 2, 3. **a** Fragment #1. **b** Fragment #2. **c** Fragment #3. **d** Fragment #1,3. **e** Fragment #1,3,2

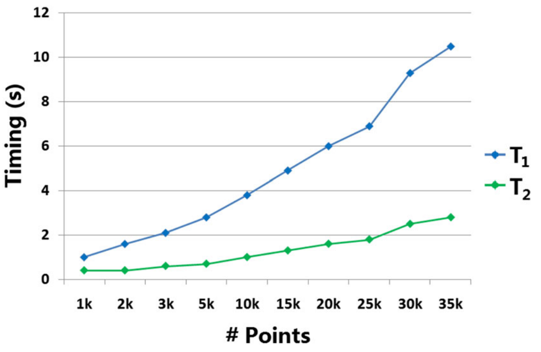


Fig. 6 Performance plot: $T1$ denotes the time cost (in seconds) for finding the longest common boundary and $T2$ denotes it for computing the alignment matrix

[solving Eq. (2)] and $T2$ denotes that for shape alignment [solving Eq. (3)]. In practice, the boundaries have hundreds to thousands of vertices and therefore 5 s suffices for finding the best matching between them.

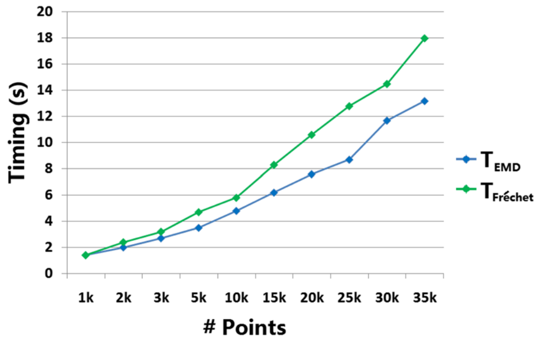


Fig. 7 Performance comparison with the Fréchet distance-based method

To compare the performance with the Fréchet distance-based method, we also apply the Fréchet distance to similarity computation. Similarly, the Fréchet distance-based similarity is defined to the length of the longest common boundary under condition that the Fréchet distance between them is within the same tolerance. Figure 7 shows the performance contrast, from which we can see that our algorithm has a conspicuous advantage. This is because our algorithm is able to quickly capture the key moments of candidate matchings, which is central to reduce the timing cost.

5.2 Robustness

Generally, the input fragments are digitalized photos. The extracted contours inevitably have noise. Therefore, a robust algorithm is desirable to deal with the practical issues. Recall that we adopt a robust curvature estimation approach that facilitates us to accurately capture the dominant geometric properties (curvatures and (R, G, B) components) of the contours, and the EMD-based technique is able to find a desirable matching configuration. Both of them are useful for the robustness of our algorithm. As Fig. 8 shows, even when the input fragments have noise along the boundaries, our algorithm can still find an accurate matching, but the Fréchet distance-based methods cannot. In fact, Fréchet distance, in its nature, depends on the one-to-one correspondence between two objects of interest, which is pretty sensitive to noise.

To quantitatively measure the accuracy of our algorithm as well as the Fréchet distance-based method, we use the respective relative errors with regard to the ground truth. Let L be the length of the real common boundary, and L_{EMD} and $L_{\text{Fréchet}}$ be respectively the similarities computed by our algorithm and the Fréchet distance-based method. Then we use $\frac{|L_{\text{EMD}}-L|}{L}$ and $\frac{|L_{\text{Fréchet}}-L|}{L}$ to measure their accuracies respectively. The detailed statistics are shown in Table 1. Experimentally, we found that the Fréchet distance-based method often captures a common segment that is shorter than the real common boundary. By contrast, our algorithm has better accuracy.

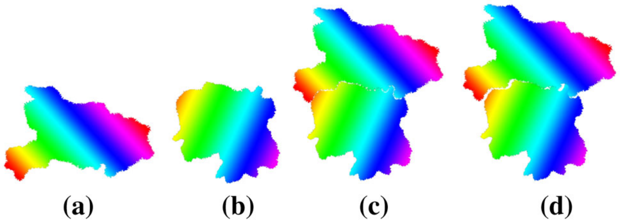


Fig. 8 When the input fragments have noise along boundaries, our algorithm can still find an accurate matching configuration, but the Fréchet distance-based methods cannot. **a** Region #1. **b** Region #2. **c** Our method. **d** Fréchet distance

Table 1 Matching accuracy

Fragments	Fréchet distance (%)	Our algorithm (%)
Figure 8	2.64	0.92
Figure 11_1	1.14	0.33
Figure 11_2	1.06	0.28
Figure 11_3	1.92	0.52
Figure 11_4	1.74	0.45
Figure 11_5	2.35	0.78

To measure the accuracy of our algorithm as well as the Fréchet distance-based method, we use the respective relative errors with regard to the ground truth (Note that the exact common boundary is known)

6 Applications

In this section, we demonstrate the uses of our algorithm to two applications, map-piece assembly and relic restoration. We finally point out a failure case as well.

6.1 Fragment assembly

Jigsaw puzzle [45] is one popular game well known all over the world. Generally, the constituent pieces usually have not only geometrical shape cues, but also visual cues such as texture and color. (Note that regular pieces with repeated visual patterns are not in our research scope.) Suppose that a Chinese map puzzle (Mainland China) requires users to assemble totally 26 individual pieces into a complete map, as shown in Fig. 9. Based on the discussion in Sect. 4, we can extract the contours first, analyze the similarity between any pair of pieces, and finally assemble them into one complete map using a greedy algorithmic scheme. The input regions are often represented by low-resolution images with inaccurate boundaries, which is a major challenge to the assembling algorithm. In spite of this, our algorithm is able to accomplish the complicated task. Figure 9 demonstrates the final assembled result. In implementation, the 26 regions have 34731 vertices totally and the required timing cost is 12.4 s.

Previous research works show that when the number of pieces becomes large, e.g., larger than 10, the assembling task will be very hard. In Fig. 10, we provide the ground truth of the jigsaw puzzle, the assembling result by our method and that of the critical point-based approach [45]. Taking Fig. 10c as an example, the critical point-based approach works well at the very beginning except that the matching configurations are not so accurate. When it is the turn is to assemble the yellow piece, overlapping cannot be avoided, which will cause a failure. In other words, even if their approach assembles the pieces in correct order, it becomes more and more difficult to find a desirable matching configuration due to the inaccurate matching configurations before. As a contrast, our method is able to give a desirable matching configuration, but relaxes the accuracy requirement of matching and thus can be applied to the scenario of quite a lot of pieces.

To further show the robustness of our algorithm, we performed more experiments. Figure 11 shows five examples, each of which is a subregion of the world map. The left column shows the input fragments, the middle column shows

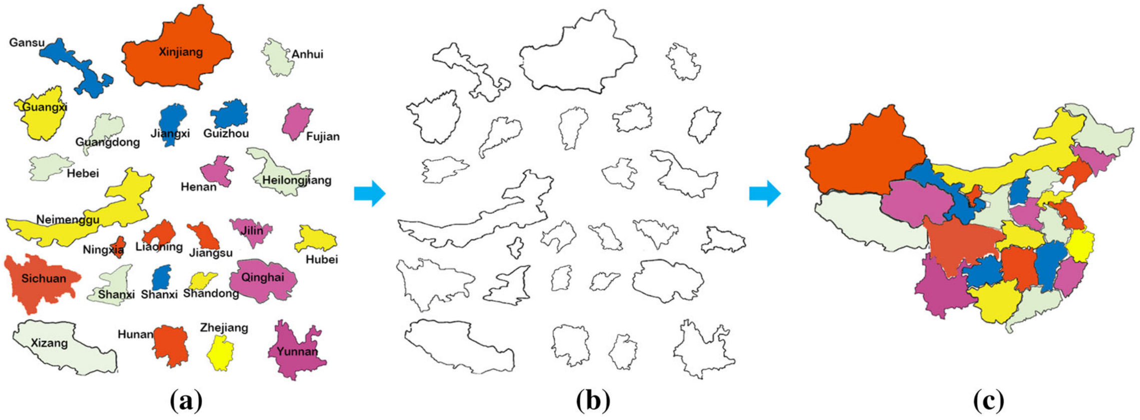


Fig. 9 Chinese map (Mainland China): when the input is a collection of separate regions (**a**), we can extract their contours (**b**) and then assemble them into a complete map based on geometric features (**c**)

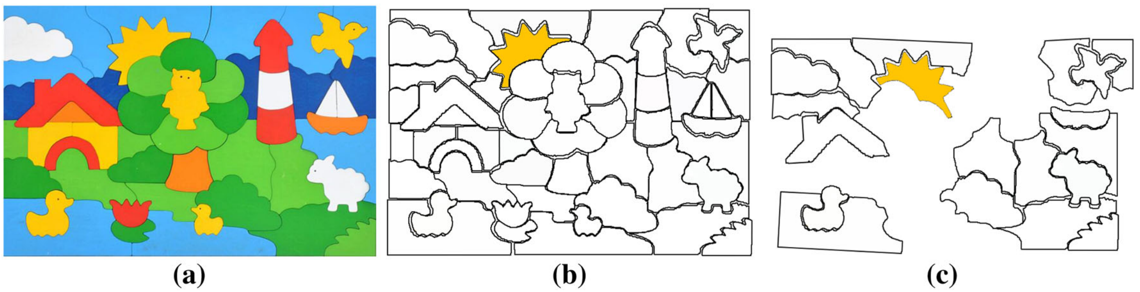


Fig. 10 The result of the jigsaw puzzle: **a** the assembling ground truth, **b** the result of the jigsaw puzzle with our method, **c** the result of the jigsaw puzzle with isthmus critical points

the similarity matrix, and the right column shows the assembled map. Note that red means that the common boundary is long, as opposed to blue.

Experimental results also show that the key to obtain a meaningful assembly lies in the similarity estimation. The advantage of our assembling algorithm comes from two aspects: (1) we adopt a robust curvature estimation approach that enables us to accurately capture the salient geometric properties and (2) the EMD technique takes into account the length correspondence and property correspondence at the same time.

6.2 Relic restoration

Cultural heritages are in urgent need of conservation and restoration. On one hand, we need to take positive measures to increase their life spans and avoid deterioration. On the other hand, digitalized restoration is also an interesting strategy [7]. Suppose that an ancient plate is broken into pieces, as Fig. 12 shows. Before archeologists assemble them by hand, we can pre-assemble the image-based constituents. First, we extract the contour of each piece (see Fig. 12b). Second, we find a reasonable arrangement so that all pieces are assembled together (see Fig. 12c). Finally, a composite image is generated (see Fig. 12d). For this example, the required time cost (boundary extraction included) is 5.3 s. It greatly relieves archeologists from tedious labour.

6.3 Failure case

However, we must point out that there are some failure cases that we cannot deal with yet. The algorithm, in its current form, assumes that the input pieces are of irregular shape, which actually implies the uniqueness of the best matching configuration. In the case where the fragments have symmetric structures, our algorithm possibly fails to give a global matching solution. For instance, if the common boundary is symmetric as Fig. 13 shows, the matching configurations between the two pieces are not unique—a wrong matching guess will lead to a failure in assembling successive fragments, since our assembling algorithm is of a greedy style.

7 Conclusions and future work

In this paper, we use partial EMD to characterize the similarity between two fragments and further propose a greedy strategy for fragment assembly. The length correspondence

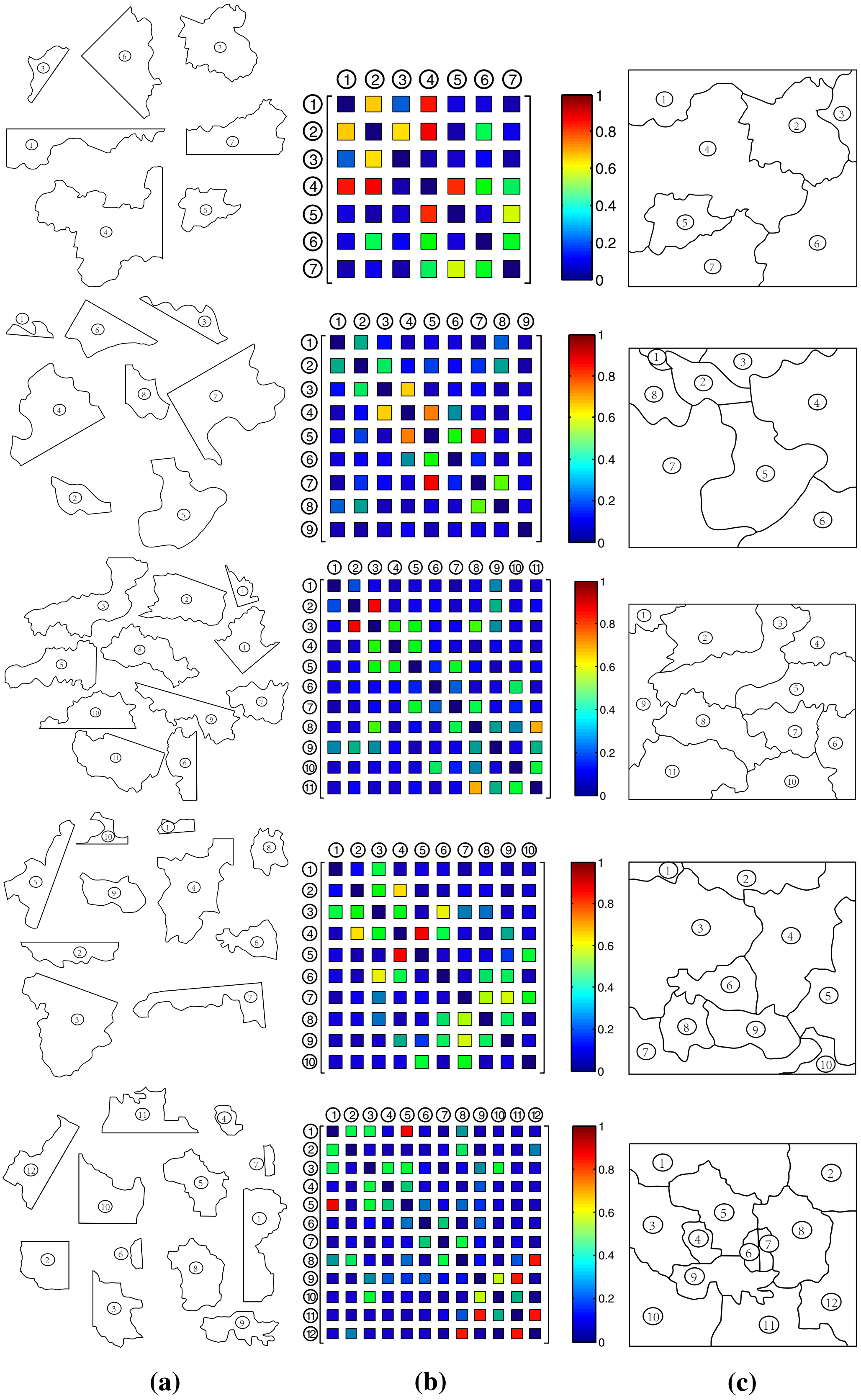


Fig. 11 More map assembling examples. **a** Inputs. **b** Similarity matrices. **c** Assembly results

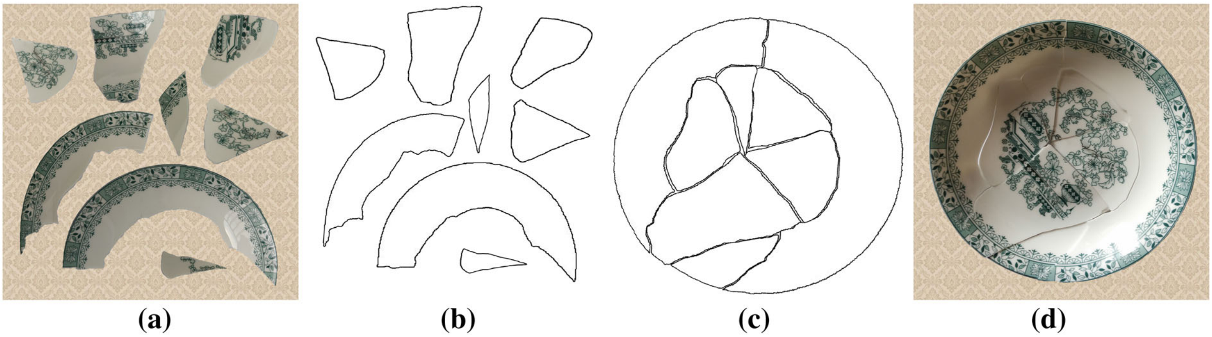


Fig. 12 Relic restoration: **a** plate fragments, **b** contour extraction, **c** contour matching, and **d** restored plate

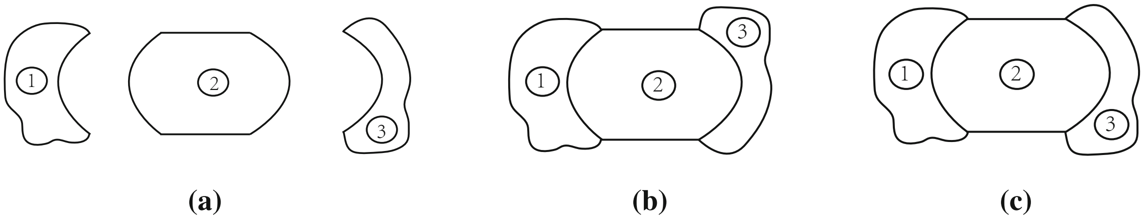


Fig. 13 Failure case: When the common boundary between two fragments (**a**) has symmetric structures that are generally generated artificially, the assembly plans may not be unique; see **b** and **c**. **a** Input, **b** Assembly plan #1 and, **c** Assembly plan #2

and the property correspondence are considered at the same when defining EMD so that we can find a better matching configuration than previous algorithms. Experimental results show that our approach is robust to noise and can be applied to the scenario of quite a lot of pieces. We believe that the approach has a great potential in many computer graphics/vision applications.

In future, we will extend our algorithm to 3D fragment assembly. Also, further speedup techniques need to be developed.

Acknowledgments We are grateful to the editors and anonymous reviewers for their insightful comments and suggestions. This work is supported by NSF of China (61300168, 61571247, 11226328), NSF of Zhejiang (LZ16F030001, LY13F020018), the Open Research Fund of Zhejiang First-foremost Key Subject (XKXL1521, XKXL1406, XKXL1429), and the International Science and Technology Cooperation Project of Zhejiang (2013C24027).

References

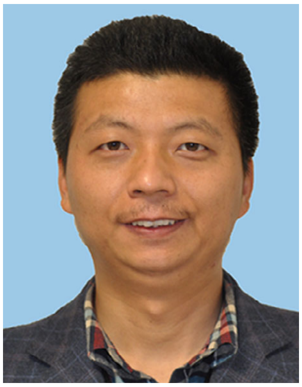
1. Agathos, A., Pratikakis, I., Papadakis, P., Perantonis, S., Azariadis, P., Sapidis, N.S.: 3d articulated object retrieval using a graph-based representation. Visual Comput. **26**(10), 1301–1319 (2009)
2. Alt, H., Buchin, M.: Can we compute the similarity between surfaces? Discret. Comput. Geom. **43**, 78–99 (2007)
3. Alt, H., Guibas, L.J.: Discrete geometric shapes: matching, interpolation, and approximation. Handb. Comput. Geom. **1**, 121–153 (1999)
4. Alt, H., Knauer, C., Wenk, C.: Comparison of distance measures for planar curves. Algorithmica **38**(1), 45–58 (2004)
5. Altantsetseg, E., Matsuyama, K., Konno, K.: Pairwise matching of 3d fragments using fast fourier transform. Visual Comput. **30**(6–8), 929–938 (2014)
6. Ancuti, C., Ancuti, C.O., Bekaert, P.: An efficient two steps algorithm for wide baseline image matching. Visual Comput. **25**(5–7), 677–686 (2009)
7. Andaló, F.A., Carneiro, G., Taubin, G., Goldenstein, S., Velho, L.: Automatic reconstruction of ancient portuguese tile panels. IEEE Comput. Graphics Appl. (2016) **(accepted)**
8. Baxter, L.A., Harche, F.: Note: on the greedy algorithm for optimal assembly. Naval Res. Logistics **39**, 833–837 (1992)
9. Buchin, K., Buchin, M., Wang, Y.: Exact algorithms for partial curve matching via the fréchet distance. In: ACM-SIAM symposium on discrete algorithms, pp. 645–654 (2009)
10. Chen, B., Pan, X.: Geodesic Fourier descriptor for 2D shape matching. In: International Conference on Embedded Software and Systems Symposia, pp. 447–452 (2008)
11. Cui, M., Femiani, J., Hu, J., Wonka, P., Razdan, A.: Curve matching for open 2D curves. Pattern Recogn. Lett. **30**(1), 1–10 (2009)
12. Cui, M., Wonka, P., Razdan, A., Hu, J.: A new image registration scheme based on curvature scale space curve matching. Visual Comput. **23**(8), 607–618 (2007)
13. Domokos, C., Kato, Z.: Realigning 2d and 3d object fragments without correspondences. IEEE Trans. Pattern Anal. Mach. Intell. **38**(1), 1–1 (2016)
14. Driemel, A., Har-Peled, S.: Jaywalking your dog—computing the Fréchet distance with shortcuts. SIAM J. Comput. **42**(5), 1830–1866 (2013)
15. Dyken, C., Dæhlen, M., Sevaldrud, T.: Simultaneous curve simplification. J. Geogr. Syst. **11**(11), 273–289 (2009)
16. Freeman, H., Garder, L.: Apictorial jigsaw puzzles: the computer solution of a problem in pattern recognition. IEEE Trans. Electron. Comput. **13**(2), 118–127 (1964)
17. da Gama Leito, H.C., Stolfi, J.: Automatic reassembly of irregular fragments. Univ. of Campinas, Tech. Rep. IC-98-06 (1998)
18. Giguere, M.: Three-dimensional puzzle assembly. US Patent 6015150

19. Goldberg, D., Malon, C., Bern, M.: A global approach to automatic solution of jigsaw puzzles. In: Conf Computational Geometry, pp. 82–87 (2002)
20. Grauman, K., Darrell, T.: Fast contour matching using approximate earth mover's distance. In: Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 1, pp. I–220–I–227 (2004)
21. Gelfand, N., Pottmann, H. Flöry, S., Hofer, M.: Reassembling fractured objects by geometric matching, ACM Trans. Graphics. (3), 569–578 (2006)
22. Huang, Z., Cohen, F.S.: Affine-invariant B-spline moments for curve matching. IEEE Trans. Image Process. **5**(10), 1473–1480 (1996)
23. James, G.M.: Curve alignment by moments. Ann. Appl. Stat. **1**(2), 2007 (2008)
24. Kanezaki, A., Harada, T., Kuniyoshi, Y.: Partial matching of real textured 3d objects using color cubic higher-order local autocorrelation features. Visual Comput. **26**(10), 1269–1281 (2010)
25. Khan, M.S., Ayob, A.F.M., Isaacs, A., Ray, T.: A novel evolutionary approach for 2D shape matching based on B-spline modeling. In: IEEE Congress on Evolutionary Computation (CEC), pp. 655–661 (2011)
26. Latecki, L.J., Megalooikonomou, V., Wang, Q., Yu, D.: An elastic partial shape matching technique. Pattern Recogn. **40**(11), 3069–3080 (2007)
27. Liu, H., Latecki, L.J., Liu, W.: A unified curvature definition for regular, polygonal, and digital planar curves. Int. J. Comput. Vision **80**(1), 104–124 (2008)
28. Maheshwari, A., Sack, J.R., Shahbaz, K., Zarrabi-Zadeh, H.: Improved algorithms for partial curve matching. Algorithmica **69**(3), 641–657 (2014)
29. Mccreath, E.: Partial matching of planar polygons under translation and rotation. In: Canadian Conference on Computational Geometry (2008)
30. Miller, J.M., Hoffman, R.L.: Automatic assembly planning with fasteners. In: IEEE International Conference on Robotics and Automation, pp. 69–74 (1989)
31. Min, G.C., Fleck, M.M., Forsyth, D.A.: Jigsaw puzzle solver using shape and color. In: The Fourth International Conference on Signal Processing Proceedings, pp. 877–880 (1998)
32. Gu, P., Yan, X.: CAD-directed automatic assembly sequence planning. Int. J. Prod. Res. **33**(11), 3069–3100 (1995)
33. Pal, A., Shanmugasundaram, K., Memon, N.: Automated reassembly of fragmented images. In: International Conference on Multimedia and Expo, pp. 625–628 (2003)
34. Parikh, D., Sukthankar, R., Chen, T., Chen, M.: Feature-based part retrieval for interactive 3d reassembly. In: IEEE Winter Conference on Applications of Computer Vision, pp. 14–14 (2007)
35. Porrill, J., Pollard, S.: Curve matching and stereo calibration. Image Vis. Comput. **9**(1), 45–50 (1991)
36. Richter, F., Ries, C.X., Cebron, N., Lienhart, R.: Learning to reassemble shredded documents. IEEE Trans. Multimedia **15**(3), 582–593 (2013)
37. Rubner, Y., Tomasi, C.: Perceptual metrics for image database navigation. Springer International **594** (1999)
38. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. Comput. Vis. **40**(2), 99–121 (2000)
39. Shirdhonkar, S., Jacobs, D.W.: Approximate earth movers distance in linear time. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)
40. Shu, X., Wu, X.J.: A novel contour descriptor for 2D shape matching and its application to image retrieval. Image Vis. Comput. **29**(4), 286–294 (2011)
41. Shuralyov, D., Stuerzlinger, W.: A 3D desktop puzzle assembly system. In: 2011 IEEE Symposium on 3D User Interfaces (3DUI), pp. 139–140 (2011)
42. Song, Y., Jin, S.: Matching sequences of salient contour points characterized by voronoi region features. Visual Comput. **28**(5), 475–491 (2012)
43. Wang, J., Yu, Z., Zhang, W., Wei, M., Tan, C., Dai, N., Zhang, X.: Robust reconstruction of 2D curves from scattered noisy point data. Comput. Aided Des. **50**(3), 27–40 (2014)
44. Wang, X., Hu, J., Zhang, D., Qin, H.: Efficient emd and hilbert spectra computation for 3d geometry processing and analysis via space-filling curve. Visual Comput. **31**(6–8), 1135–1145 (2015)
45. Webster, R.W., Lafollette, P.S., Stafford, R.L.: Isthmus critical points for solving jigsaw puzzles in computer vision. IEEE Trans. Syst. Man Cybern. **21**(5), 1271–1278 (1991)
46. Wei, G., Xiao-dong, S., Huan-ling, L.: Automatic assembly location method based on particle filter. Comput. Integr. Manuf. Syst. **20**(7), 1615–1624 (2014)
47. Xu, C., Liu, J., Tang, X.: 2D shape matching by contour flexibility. IEEE Trans. Pattern Anal. Mach. Intell. **31**(1), 180–186 (2009)
48. Zheng, Y.F., Pei, R., Chen, C.: Strategies for automatic assembly of deformable objects. In: IEEE International Conference on Robotics and Automation, pp. 2598–2603 (1991)

Meng Zhang is currently a postgraduate in the Faculty of Electrical Engineering and Computer Science at Ningbo University. He participated in a research project of calibration of multiple Kinects.

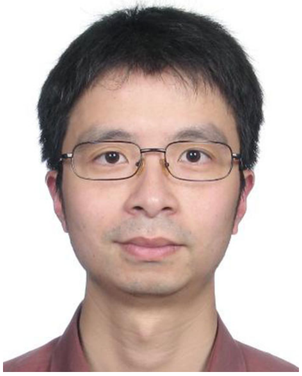
Shuangmin Chen got her master's degree in 2007 at the Zhejiang University, China. She is now working as a lecturer at the Faculty of Electrical Engineering and Computer Science, Ningbo University. Her research interests include machine learning and digital geometry processing. She has published 15 papers in international conferences or journals.

Zhenyu Shu got his Ph.D. degree in 2010 at the Zhejiang University, China. He is now working as an associate professor at Ningbo Institute of Technology, Zhejiang University. His research interests include computer graphics, digital geometry processing and machine learning. He has published over 30 papers in international conferences or journals.



Shi-Qing Xin is an associate professor at the Faculty of Electrical Engineering and Computer Science at Ningbo University. He received his Ph.D. degree in applied mathematics at Zhejiang University in 2009. His research interests include computer graphics, computational geometry and 3D printing.

Jieyu Zhao is a full professor in the Faculty of Electrical Engineering and Computer Science at Ningbo University. He received his Ph.D. degree in London University in 1995 and majored in artificial neural networks. His research interests include artificial intelligence and machine learning.



Guang Jin is a full professor in the Faculty of Electrical Engineering and Computer Science at Ningbo University. He received his Ph.D. degree at Zhejiang University. His research interests include mobile computing and computer networks.

Rong Zhang received her Ph.D. degree in communication and information system from Ningbo University in 2015. She is currently an associate professor at Ningbo University, China. Her main interests are in digital image forensics and digital watermarking.

Jürgen Beyerer received his Diploma and Doctoral degree in engineering from Universität Karlsruhe in 1989 and 1994, respectively. He is a full Professor for Interactive Real Time Systems at the KIT Karlsruhe and the Head of the Fraunhofer-Institute of Optronics, System Technologies and Image Exploitation IOSB. Her research interests include automated visual inspection, optimal acquisition of image data, statistical signal theory, fusion of data and information from heterogeneous sources, active vision, and system theory.