Vol.2013-CVIM-187 No.51
2013/5/31

A Linear Algebraic Approach to Intraclass Shape Analysis and Its Application in Archaeological Research

Min Lu[1,a)] Bo Zheng[1,b)] Jun Takamatsu[2,c)] Ko Nishino[3,d)] Katsushi Ikeuchi[1,e)]

Abstract: Shape analysis is one of the fundamental problems in computer vision. In particular, if objective shapes belong to the same class, the problem is specified to what is named as intraclass analysis. In this paper, we focus on the general problem of intraclass shape analysis. A novel framework for analyzing similar shapes is proposed. Given samples within a certain category, we first manage to reveal hidden structural information with low-rank matrix recovery theory, and further develop effective algorithms under the guidance of this information for several related topics, including intraclass shape restoration, clustering and comparison. All these processes are fully automatic, without using any prior knowledge about the given shapes as well. Experimental results verify the feasibility and effectiveness of our proposed method.

1. Introduction

Shape analysis is one of the most fundamental problems in computer vision and an elementary step to artificial intelligence. There are several sub-topics related to shape analysis, including salient region detection, geometry comparison, object recognition, etc. In this sequence of questing a better way to make intelligent systems, researchers are inspired by the cognition pattern of ourselves. Seeming to be almost naturally capable of observe and further analyze the creatures and creations, human start his cognitive process from shape analysis.

As a relatively wide topic, some studies about shape analysis focus on analyzing a single object only. For example, to detect feature point or salient region via curvature analysis, or to extract feature lines such as ridges and valleys. If two or more objects are taken into consideration, the problem turns into shape comparison.

1.1 Interclass and Intraclass Problems

Generally speaking, depending on the types of target objects, the problem of shape analysis can be further divided as interclass analysis and intraclass analysis. The former issue crosses over objects without similarity prerequisite; while the latter issue focuses on the analysis on those targets belong to the same category. From our former experiences, objects from the same class usually have similar shapes and structures. Therefore finer comparison is required for intraclass analysis.

1 The University of Tokyo
2 Nara Institute of Science and Technology
3 Drexel University
a) lumin@cvl.iis.u-tokyo.ac.jp
b) zheng@cvl.iis.u-tokyo.ac.jp
c) j-taka@is.naist.jp
d) kon@drexel.edu
e) ki@cvl.iis.u-tokyo.ac.jp

Fig. 1 shows two different data sets, belonging to interclass and intraclass problems respectively. Notice that these two different kinds of problems actually stay in different levels of analysis: while in interclass analysis we pay more attention to recognition, which is to solve the problem "what is this?", intraclass analysis focus on more detailed shape comparison, e.g. "How to further divide them into sub-groups? What is the difference between them?".

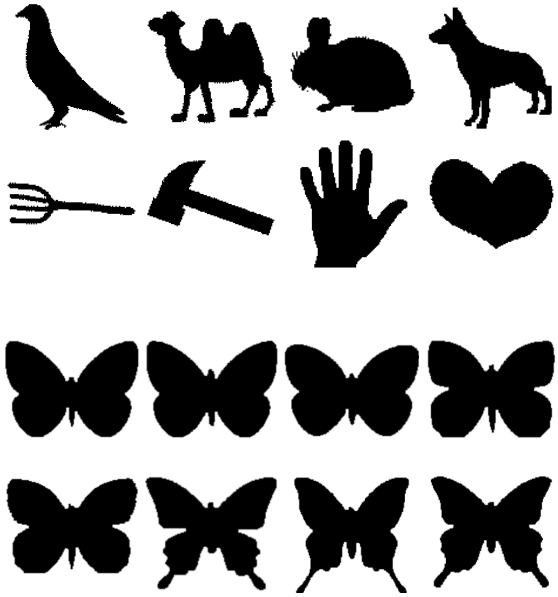


Fig. 1: Two data set of 2D silhouettes images. The top two rows constitute one data set, where samples are totally different objects from each other. This leads to an interclass problem. The rest two rows form the other data set, where samples are all butterflies, with slightly different shapes. This belongs to the intraclass analysis. These images are from a large binary image database collected by the LEMS Vision Group at Brown University.

In this paper, we focus on shape analysis with both 2D and 3D models. Here 2D data are represented as contours or silhouettes, as shown in Fig. 1; while 3D shapes are mainly triangle meshes or point clouds scanning from real world.

1.2 Motivation

Compared with general shape comparison problems, intraclass shape analysis enjoys better semantic explanation and higher processing accuracy, which bring it increasing concern with wide range of applications.

A typical usage of intraclass shape analysis is biometrics. It is used to identify individuals from their characteristics or trait. Taking 3D face recognition as an example, 3D geometry of the facial part is captured and then compared with others. Since geometry is invariable with the change of illumination or head pose, this technique overbears its 2D counterpart with significantly higher accuracy.

The application of intraclass shape analysis in manufacturing industry is also in the ascendant. Take the automobile industry as an example. For manufacturers, if same or similar parts of old car models can be reused in the design and production process of the new models, that would be both economical and efficient. Therefore 3D retrieval system for workpieces attracts widespread attention. In such a retrieval system, geometry shapes of workpiece models are acquired and archived in advance. When a query is given, the system will analyze its shape, compare it with the current samples, and then return most similar objects. This system provides increased productivity and decreased cost by facilitating optimal workpieces pickup flow and maximal reuse extent. We believe that with the development of the new 3D printing technique, intraclass shape analysis will become even more important.

Another important application of intraclass shape analysis is the protection and study on historic relics. Nowadays, three-dimensional digital replicas play an increasingly important role in cultural heritage preservation. With current 3D data acquisition technology, such as laser rangefinders, the geometric information of real-world objects can be accurately and reliably digitized. These 3D digital models can then be used for various archaeological studies. In addition, historic relics are often only partially preserved due to various causes, such as weather and vandalism. In order to avoid a great loss of cultural heritage, finding effective ways to restore the original appearance of these relics is important. We will discuss about this topic in Section 3.

Besides, intraclass shape analysis also shows a significant potential in medical diagnosis and treatment, e.g. personalized tooth shape estimation. We hope our study can contribute to the task of improving our lives.

1.3 Related Work

Shape restoration

Several approaches have been proposed for shape restoration. For the purpose of filling holes on given polygon meshes or point clouds, either mesh-based methods [30] or volumetric approaches [16], [26], [36] can be used to achieve smooth continuation by imposed localized geometric constraints. This kind of completion methods is suitable for filling holes, but in case large region or structural information is lost, hole-filling methods will be no longer suitable. An alternative strategy using geometric priors can be employed. Usually patches with similar surface characteristics are selected from either the incomplete object itself, such as [7], [43], [50], or analogous candidate models, including [29], [37]. Notice that all these methods handle a single sample at a time.

The studies about reconstruction from depth video, such as [38], as well as markerless motion capture systems also suffer from data incompletion. By making use of the correspondences between frames, and acquiring accumulate geometric information over time, the restoration problem can be solved.

Intrinsic shape analysis

A recent hot topic in shape analysis is the intrinsic subproblem, where only isometric deformation is taken into consideration. For example, the eigenfunctions of Laplace-Beltrami operator shows its power to deal with isometric deformation in [11], [40]. A study about analyzing deformable two-dimensional shapes is introduced in [9]. Based on *Gromov-Hausdorff* distance and intrinsic geometric properties of shapes, they present deformation-invariant shape comparison method.

Notice that these studies belong to intraclass shape analysis as well. However, the scope of research objects are different from our study in this paper— while the intrinsic analysis focuses on the same object under different isometric postures, we try to handle the comparison between a group of similar but different objects, which are ideally under the same posture.

Partial shape matching

A regularized partial matching method of rigid shapes is discussed in [8], and a partial similarity detection method using the notion of Pareto optimality is introduced in [10]. Moreover, with local descriptors, such as *shape contexts* introduced in [2] and *spin images* discussed in [24], as well as effective indexing algorithms, many partial matching methods by shape similarity are proposed, such as the studies in [3], [22].

Statistical shape analysis

Another branch of methods to analyze a set of shapes is statistical shape analysis discussed in [18], where statistics are measured to describe geometrical properties from similar shapes or different groups. Usually principal component analysis (PCA) [25] is used to analyze the shape variability. As for the identifiable region detection problem, researches on sparse subspace clustering such as [19], as well as the biclustering [14] provide the theoretical basis.

Cluster culture relics

Extensive studies of the Bayon faces have been conducted

from the cultural and archaeological perspectives. A groundbreaking study on the classification of Bayon faces was done by the Japanese Government Team for Safeguarding Angkor (JSA) in [35]. Based on the observation and analysis from their experienced experts, those faces were roughly divided into three groups: *Deva*, *Devata* and *Asura*, meaning *god*, *goddess* and *devil* respectively. Meanwhile, several more objective methods of classification for comparison were proposed. For example, a framework for clustering images of face carvings at archaeological sites was presented in [28], where the pairwise similarities were computed from local facial features (eyes, nose, etc.). This method was effective when dealing with the Devata goddesses depicted in *Angkor Wat*. And a classification study about Bayon faces with several different methods was established by [27], using depth images generated from scanned 3D data. Other related work includes tracing the sequence of constructing the Bayon by [15], and describing the deterioration of sandstone blocks in the temple by [45].

1.4 Contributions

In this paper, we concentrate on the problem of intraclass shape analysis, including restoration, clustering and comparison. We present novel algorithms for all stages of this process.

Intraclass shape analysis with $\mathcal{L}_1$ minimization

We address the problem of restoring a group of similar objects with damaged samples, and propose a restoration method that builds upon the *low-rank matrix recovery* theory to restore all incomplete samples simultaneously. To the best of our knowledge, our work is the first attempt to solve shape restoration problem from the algebraic viewpoint. A general and elegant framework is proposed, without using any prior knowledge about the shape. We further extend the original matrix recovery theory to more complex case where large missing areas can be handled as well.

Joint shape restoration and cluster analysis

We introduce a two-step shape recovery strategy, which combines restoration and cluster analysis together, to further refine the restoration result and divide the data set into subcategories. Due to the fact that we use unsupervised classification method, pre-labeled data is no longer needed to separate the data set. Notice that our method is based on objective measurements and comparison as well. Experimental results with real world culture heritages demonstrate the effectiveness of our proposed method.

A quantitative intraclass shape comparison method

We present a quantitative method to compare similar shapes, extending shape similarity evaluation with deformation field. A tensor decomposition based approach to the analysis of similar shapes, and an reinforced scheme to detect identifiable regions are also proposed. These are novel and interesting attempts to analyze similar objects.

2. Preliminaries

Before we start exploring the world of intraclass shape analysis, a first look at the *low-rank matrix recovery* theory is necessary, since it is the theoretical basis of our work. We would like to give a brief introduction about this theory in this preliminary section.

2.1 Norm

Let us start from the norm. Its mathematical meaning is clear: the norm of an object, such as vectors, is a quantity that describes the length or size of this object. The $\mathcal{L}_p$ norm in general case is defined as:

$$\|x\|_{\mathcal{L}_p} = (\sum_i |x_i|^p)^{1/p}, \tag{1}$$

where $x \in \mathbb{R}^n$, $p \in \mathbb{R}$, $p \geqslant 1$.

From this definition, we can derive the most common two norms— *taxicab* ($\mathcal{L}_1$) norm and *Euclidean* ($\mathcal{L}_2$) norm, as shown below respectively:

$$\|x\|_{\mathcal{L}_1} = \sum_i |x_i|, \tag{2}$$

$$\|x\|_{\mathcal{L}_2} = \sqrt{\sum_i |x_i|^2}. \tag{3}$$

In particular, if we let $p = 0$, then we get the "$\mathcal{L}_0$ norm":

$$\|x\|_{\mathcal{L}_0} = \sum_i |x_i|^0, \tag{4}$$

where $0^0 \doteq 0$. Notice that $\mathcal{L}_0$ norm is special, since it is actually not a norm in the usual sense. However, due to the fact that $\mathcal{L}_0$ norm counts the number of non-zero entries, it is widely used to evaluate whether a mathematical object is sparse or not, such as the study in [34].

2.2 Low-Rank Matrix Recovery

Low-rank matrix recovery, which is introduced in [12], [48], [49], is also known as *robust principal component analysis* (Robust PCA). The essential idea of this theory is to recover corrupted entries of a matrix using structural information of the matrix itself. An example is shown in Fig. 2.

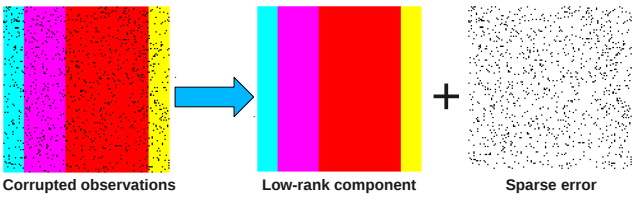


Fig. 2: An illustration of low-rank matrix recovery. The input corrupted observation matrix can be separated into two components: an underlying low-rank matrix and a sparse error matrix.

Given the observation matrix $D \in \mathbb{R}^{m \times n}$, which is generated by corrupting some entries of an unknown low-rank matrix, we aim to find a proper decomposition, that separates it into two meaningful parts:

$$D = A + E. \tag{5}$$

Here $A \in \mathbb{R}^{m\times n}$ is a low-rank matrix, representing the original data without corruptions; while $E \in \mathbb{R}^{m\times n}$ is the error matrix, which is supposed to be sparse.

Robust principal component analysis solves this problem by seeking a possible A with the lowest rank that could have generated the observation D, while subjecting the error matrix E to a sparseness constraint: $\|E\|_0 \leqslant k$. Here the $\mathcal{L}_0$ norm is employed to measure the matrix sparseness. Thus the initial decomposition problem is turned into an optimization:

$$\min_{A,E} \quad \text{rank}(A) + \gamma\|E\|_0, \quad s.t. \quad A + E = D, \tag{6}$$

where γ is the weighting parameter that trades off the rank of the solution and the sparseness of the error.

Unfortunately, the optimization problem (6) is highly non-convex, and currently with no efficient solution. A tractable optimization, however, can be obtained by relaxing the original problem. By replacing the $\mathcal{L}_0$ norm with the $\mathcal{L}_1$ norm, and by measuring the rank with the nuclear norm $\|A\|_*$, problem (6) can be converted to a tractable convex optimization:

$$\min_{A,E} \quad \|A\|_* + \lambda\|E\|_1. \quad s.t. \quad A + E = D. \tag{7}$$

Here $\|A\|_*$ is the nuclear norm of matrix A, defined as the sum of its singular values: $\|A\|_* \doteq \sum_i \sigma_i(A)$. $\|E\|_1$ refers to the L_1 norm of matrix E and the weighting parameter λ is in the form $c/\sqrt{m}$, where c is a constant, and typically set to be around 1. Notice that the new objective function in problem (7) is continuous and convex, so it can be solved efficiently via convex optimization, such as the *augmented Lagrange multipliers* (ALM) algorithm.

Compared to ordinary *principal component analysis* (PCA), this method is more robust to outlying and corrupted observations. Therefore it can be used in various complex problems, such as batch image alignment and robust photometric stereo, etc.

3. Intraclass Shape Restoration

Given a group of similar shapes, before any further comparison, the integrity of the data should be checked first. In real world cases, it will not be that surprising if the shape information we obtained is incomplete.

A variety of different reasons may cause this problem. Sometimes it is because of the limitations of data acquisition method. For instance, laser scanning usually fails in case of self-occlusion or some special surface reflection properties. Another kind of reason is that the data sources are different. Take the dental simulation as an example. While some recently techniques, e.g., cone beam computed tomography (commonly referred to by the acronym **CBCT**), can provide the full geometry of our teeth including root portions, data acquired via traditional casting method will contain the crown parts only. Moreover, source objects may be incomplete as well, especially in the case of cultural relics. Due to natural and human factors, e.g., weathering and vandalism, historic cultural relics are often partially damaged. An example is demonstrated in Fig. 4(d).

So here comes the question, can we restore the incomplete samples by making use of the complete ones? This is a problem of practical significance. Let us take the dental simulation as an example again, since the latest medical equipments are usually both expensive and not popular, patients will benefit from this shape restoration algorithm to avoid extra medical expenses.

3.1 Framework Overview

Given a group of similar objects in a uniform posture, including partially damaged and incomplete samples, we propose a novel algorithm which automatically restores all these objects simultaneously. Aiming to make use of the group shape similarity, we formulate the shape restoration task as a low-rank matrix recovery problem: samples are first represented as fixed-length vectors via shape matching, and then restored to their original shapes by adopting matrix recovery, which can be solved effectively via convex optimization discussed in [6], [13]. Fig. 3 depicts an overview of our proposed method.

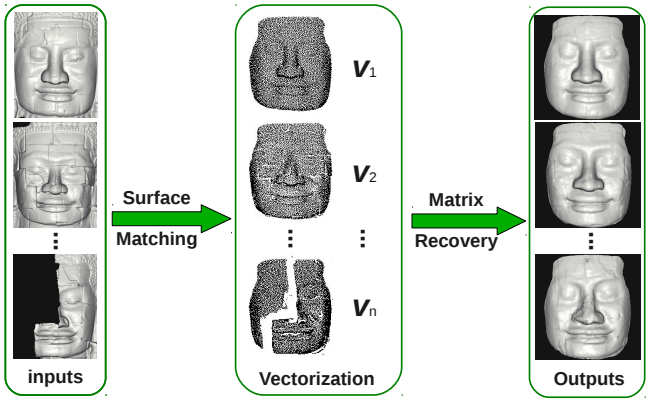


Fig. 3: the pipeline of our shape restoration method. Input shapes are first aligned and dense correspondences are acquired using a surface matching scheme. By stacking coordinates of these corresponding points, input shapes are represented as fixed-length vectors and then a matrix recovery procedure is used to accomplish the restoration.

3.2 Shape Matching and Representation

Before adopting our restoration algorithm, the shape representation has to be reformed. Each sample should be properly represented as a fixed-length vector based on accurate correspondences. We will describe this procedure in this section.

3.2.1 Preprocessing

Given the data set, denoted as $\{S_i\}_{i=1}^N$, we first unify the scale via *Procrustes analysis*, which is introduced in [23]. Here the *root mean square distance* (RMSD) from all points to its mass center is used for scaling:

$$s = \sqrt{\frac{\sum_i^K \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}{K}}, \tag{8}$$

where K is the number of points on a certain object. Notice that this may not be accurate enough in case that a relatively large part is missing. Therefore a certain degree of manual correction

is necessary.

Then we rigidly align these object together. Considering about the possible data corruption, we manually select a relatively complete sample from the whole data set, denoted as S_T. This one will serve as the template, to which all the other samples will be rigidly aligned in a preprocessing phase via *iterative closest point* (ICP) algorithm introduced in [4]. In particular, if target objects are reflection symmetrical, e.g. human faces, the axis (2D case) or plane (3D case) of symmetry can be used to help the rigid alignment processing. Here we utilize an effective reflection symmetry detection method introduced in [47].

After the above preprocessing, all samples are rigidly aligned and scaled to an unified size. A representative point set P_{S_T} with an adequate number of points is extracted from the region of interest (ROI) on the template S_T. An example is shown in Fig. 4(c). This can be done by adopting an uniform sampling.

3.2.2 Shape matching

In the whole process of shape restoration via matrix recovery, a crucial step is to properly represent the shape of each sample using a fixed-length vector, so that accurate correspondences are established among all input objects.

Given a data set of similar shapes, we would like to first establish dense correspondences among this group for further analysis. Actually the problem how to find a meaningful and accurate correspondence between two or more shapes itself is a fundamental task for shape analysis. A number of different methods have been proposed in this field, such as [32], [51], [52], [53]. A comprehensive survey about shape correspondence is given in [46] for further reading.

In our work, since all the samples belong to the same class, which means that they are quite similar to each other, we choose a deformation-driven approach to acquire correspondences. After that, the desired correspondences between all samples $\{S_i\}_{i=1}^N$ can be approximately obtained via *nearest neighbor searching*. Notice that here a distance threshold is set: if there is no point within this threshold, correspondence is marked as a null point. The above matching process is shown as Algorithm 1.

Algorithm 1 Obtaining Correspondences via Shape Matching

Input: $\{S_i\}_{i=1}^N$, S_T and P_{S_T}

Preprocessing: rigidly align $\{S_i\}_{i=1}^N$ to S_T

for $i = 1 \rightarrow N$ **do**

// Deform S_T to $\widehat{S_T}$ that approximates the target S_i

$\widehat{S_T} = \mathcal{F}_{S_T \rightarrow S_i}(S_T)$

Update the point set on template S_T: $P_{S_T} \rightarrow \widehat{P^i_{S_T}}$

//for $\widehat{P^i_{S_T}}$, search for the nearest-neighbors on target S_i

$P_{S_i} = \mathcal{F}_{NN}(\widehat{P^i_{S_T}}, S_i)$

end for

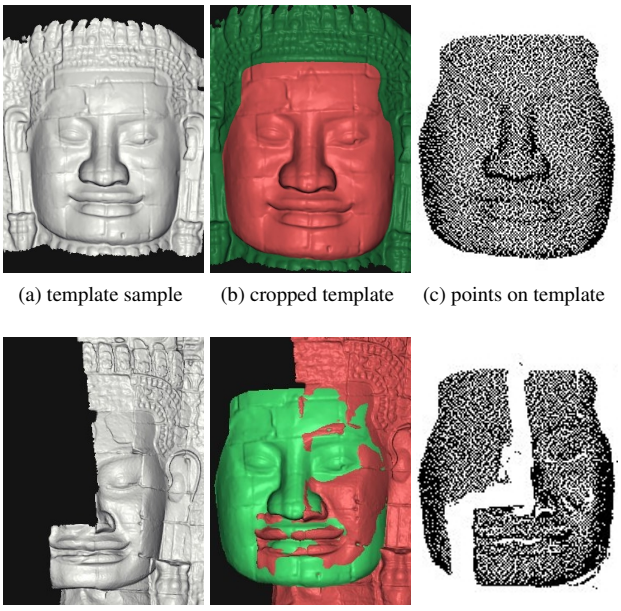
Output: $\{P_{S_i}\}_{i=1}^N$

As for the shape deformation phase, a *moving least squares* (MLS) deformation discussed in [1], [41] is employed. N pairs of control points, denoted as $\{\mathbf{p}_i\}_{i=1}^N$ and $\{\mathbf{q}_i\}_{i=1}^N$, are manually selected as sparse correspondences. Here $\{\mathbf{p}_i\}_{i=1}^N \in \mathbb{R}^3$ are the original positions on source model S_0, and $\{\mathbf{q}_i\}_{i=1}^N \in \mathbb{R}^3$ are the corresponding deformed positions on destination shape S_d. For an arbitrary point $\mathbf{x} \in \mathbb{R}^3$ on the source model S_0, let $F_\mathbf{x}$ denote the transformation that gives the corresponding position of point $\mathbf{x}$ on S_d after the deformation. According to the MLS theory, $F_\mathbf{x}$ could be determined by solving the following optimization:

$$\min_{F_\mathbf{x}} \sum_{i=1}^{N} \frac{1}{d(\mathbf{x}, \mathbf{p}_i)^{2\alpha}} \|F_\mathbf{x}(\mathbf{p}_i) - \mathbf{q}_i\|_2, \tag{9}$$

where $d(\mathbf{x}, \mathbf{p}_i)$ is the distance between $\mathbf{x}$ and $\mathbf{p}_i$, α is a system parameter.

In order to get better deformation results, geodesic distances are used in the weight function. Given one 3D shape represented by a triangle mesh, the geodesic distance between two points on its surface can be approximated with the length of the shortest path from one to the other, which can be calculated by *Dijkstra's algorithm* introduced in [17]. Moreover, the mapping $F_\mathbf{x}$ is assumed to be an affine transformation, consisting of a linear transformation M followed by a translation T: $F_\mathbf{x}(\mathbf{x}) = M\mathbf{x} + T$.



(a) template sample (b) cropped template (c) points on template

(d) a damaged sample (e) shape matching (f) corresponding points

Fig. 4: Establishing shape correspondence. The first column are two input shapes, where (a) is relatively complete and selected as the template, while (d) is a heavily damaged sample; (b) is a cropped template sample that keeps the region of interest only; (e) shows the matching procedure between these two shapes; (c) shows a dense set of sampling points on the template, and (f) is the corresponding points on example (d).

So far, we obtained a dense point set $\{P_{S_i}\}_{i=1}^N$ with correct correspondences for all given samples $\{S_i\}_{i=1}^N$. For a certain point cloud P_{S_i}, in order to reform it as a length-fixed vector, the (x, y, z) coordinates of all points in P_{S_i} are stacked to form a vector $\mathbf{v}_{S_i}$.

As for the vectorization of each sample, it is obvious that this procedure will generate vectors with the same length, since the number of sampling points are fixed. Notice that points corresponding to damaged parts may be marked as null points in our scheme. These null points could be substituted with nearby points on the object's convex hull or bounding box for actual calculation. Fig. 4(f) shows an example where points corresponding to the missing right side of face are chosen from the bounding box instead.

3.3 Shape Restoration via Matrix Recovery

From the experience of our daily lives, we know that for a group of similar objects, even if there is a damaged or missing part on one of them, we can still infer what does the original shape look like from other intact samples. To leverage this fact, we propose a novel method to restore a group of similar shapes based on the low-rank matrix recovery theory we introduced in Section 2.

3.3.1 Problem Formulation

Given a sufficiently large number of similar shapes from the same object category, denoted as $\mathbf{C}$, where some samples are partially damaged and incomplete, our aim is to get rid of corruption as much as possible and recover their undamaged shapes. Let $\{S_i^0\}_{i=1}^N$ and $\{S_i\}_{i=1}^N$ denote their original shapes when they were just completed and the corresponding observations, i.e., raw measured data in the real world respectively.

An assumption that $\{S_i^0\}_{i=1}^N$ are linearly correlated is required as a premise. Let $\text{span}(S_1^0, \dots, S_N^0)$ denote the linear span of these shapes:

$$\text{span}(S_1^0, \dots, S_N^0) = \{\sum_{i=1}^{n} \alpha_i S_i^0 | \alpha_1, \dots, \alpha_n \in \mathbb{R}\}, \tag{10}$$

where $\{\alpha_i\}_{i=1}^n \in \mathbb{R}$ are coefficients. This represents the intersection of all subspaces containing this set. Therefore an arbitrary sample S^0 from the same category $\mathbf{C}$ will approximately lie in this linear span:

$$S^0 \approx \sum_{i=1}^{N} \alpha_i S_i^0. \tag{11}$$

Notice that these samples, $\{S_i^0\}_{i=1}^N$, are actually no need to be linearly independent.

This assumption means the dimension of $\text{span}(S_1^0, \dots, S_N^0)$ should be much smaller compared with the superposition of N samples. In other words, after the shape matching procedure introduced in Section 3.2, $\{S_i^0\}_{i=1}^N$ can be reformed as vectors $\{\mathbf{v}_{S_i}^0\}_{i=1}^N$. The matrix of their combination, denoted as:

$$A \doteq [\mathbf{v}_{S_1}^0 | \cdots | \mathbf{v}_{S_N}^0] \in \mathbb{R}^{M \times N}, \tag{12}$$

should be of a much smaller rank than the total number of samples N. This reveals that A is approximately a low-rank matrix. In our method, this assumption of linear correlation is the only prior knowledge we rely on to restore the corrupted samples.

Similarly, an observation data matrix D can be formed by $\{\mathbf{s}_i\}_{i=1}^n$. The difference between A and D corresponds to the corruption, which is denoted as matrix E. The observation can then be decomposed as

$$D \doteq [\mathbf{v}_{S_1} | \cdots | \mathbf{v}_{S_N}] = A + E. \tag{13}$$

Usually the defect area is much smaller compared with the complete parts, which means error matrix E should be sparse and most of its entries are zero. Considering that matrix A is approximately low-rank, as analyzed earlier, the task of restoring this group of similar shapes can be formulated as a *low-rank matrix recovery* problem, as we discussed in Section 2.

3.3.2 Reformulation including Missing Data

One shortage of the shape recovery method introduced in the previous section is, missing data and damaged parts are treated in the same way, leading to inaccurate restorations. For instance, if the missing parts of one sample are considerably large, while the total number of samples are limited, which means the size of data matrices is relatively small, the corruption matrix E would be no longer sparse if the missing parts are not excluded. Inaccurate results may be generated in this case. In order to improve the restoration method to avoid this problem, cases of data loss and corruption have to be distinguished, and treated differently.

Suppose the size of matrix D in Eq. (7) is $M \times N$. Let $\Omega \subseteq [M] \times [N]$ denote the positions of observed entries of D, which is called the *support set* of D. Its complementary set, Ω^c, corresponds to those missing entries, which have been automatically localized in the preprocessing step. Let P_Ω be an orthogonal projection supported on Ω, defined as

$$P_\Omega(X) = \begin{cases} X_{ij}, & (\text{i}, \text{j}) \in \Omega, \\ 0, & (\text{i}, \text{j}) \notin \Omega, \end{cases} \tag{14}$$

where $X_{M \times N}$ is an arbitrary matrix. This projection can be used as a mask to exclude missing entries of matrix E in Eq. (7), making the rest of this difference matrix E sparse. The former optimization Eq. (7) is then modified as

$$\min_{A,E} \quad \|A\|_* + \lambda \|P_\Omega(E)\|_1, \quad s.t. \quad A + E = D. \tag{15}$$

Notice that the objective function is still convex.

We use augmented Lagrange multipliers to solve the above optimization. The procedures are outlined as Algorithm 2. Notice that in this case the augmented Lagrangian function becomes:

$$L(A, E, Y, \mu) = \|A\|_* + \lambda \|P_\Omega(E)\|_1 + \langle Y, D - A - E \rangle + \frac{\mu}{2} \|D - A - E\|_F^2. \tag{16}$$

Algorithm 2 Augmented Lagrange Multiplier for Our Problem

Input: Observation matrix $D \in \mathbb{R}^{m \times n}$, support set Ω and λ.

Initialization: $Y_0 = D/J(D)$; $E_0 = 0$; $\mu_0 > 0$; $k = 0$.

while not converged **do**

// Solve $A_{k+1} = \arg\min_A L(A, E_k, Y_k, \mu_k)$

$(U, S, V) = svd(D - E_k + \mu_k^{-1} Y_k)$;

$A_{k+1} = U\mathcal{S}_{\mu_k^{-1}}[S]V^T$.

// Solve $E_{k+1} = \arg\min_E L(A_{k+1}, E, Y_k, \mu_k)$

$E_{k+1} = P_\Omega(\mathcal{S}_{\lambda\mu_k^{-1}}[D - A_{k+1} + \mu^{-1}Y_k]) + P_{\Omega^c}(D - A_{k+1} + \mu^{-1}Y_k)$

$Y_{k+1} = Y_k + \mu_k(D - A_{k+1} - E_{k+1})$

Update μ_k to μ_{k+1}

$k \leftarrow k + 1$

end while

Output: A_k, E_k

Similarly to [31], we introduce a soft-thresholding (shrinkage) operator for convenience. It is defined as:

$$\mathcal{S}_\epsilon[x] \doteq \begin{cases} x - \epsilon & \text{if } x > \epsilon, \\ x + \epsilon & \text{if } x < -\epsilon, \\ 0 & \text{otherwise,} \end{cases} \tag{17}$$

where $x \in \mathbb{R}$ and $\epsilon > 0$. This operator can be extended to multi-dimensional arrays by applying it element-wise.

3.4 Experimental Results

Fig. 5: Example 3D shapes from the database of facial sculptures in *Bayon*. Notice that many of them are incomplete, due to natural decay and damage from vandalism.

We conducted experiments to restore the 3D shapes of real-world cultural relics. The Bayon facial statue database introduced in Appendix A is used, including 151 facial sculptures scanned from the temple *Bayon*. A part of these models are shown in Fig. 5. Due to weathering, vandalism, and some other reasons, many sculptures are incomplete, and some of them are damaged so heavily that only a small part is preserved.

Fig. 6: The small data set for verification. The top two rows: the chosen six well-preserved faces; The rest: demonstrations of synthesized samples, generated by cutting down a certain facial area. Holes are filled using smooth surfaces.

A relatively complete sample, *No*. 15N, shown in Fig. 4(a), is chosen as the template. Compared to the outer part of a facial sculpture, such as the ears and the headwear, the inner part (the face) contains more information we are interested in. Taking this into consideration, before generating dense correspondences, the outer part of the template is masked out, as illustrated in Fig. 4(b).

Besides, initially each sample contains around 500,000 points and 1,000,000 triangles. In order to reduce the amount of calculation, the template was down-sampled to 10,000 points before the shape matching. This leads to an observation matrix D with 30,000 rows—three times of the number of sampling points.

3.4.1 Verification Experiment

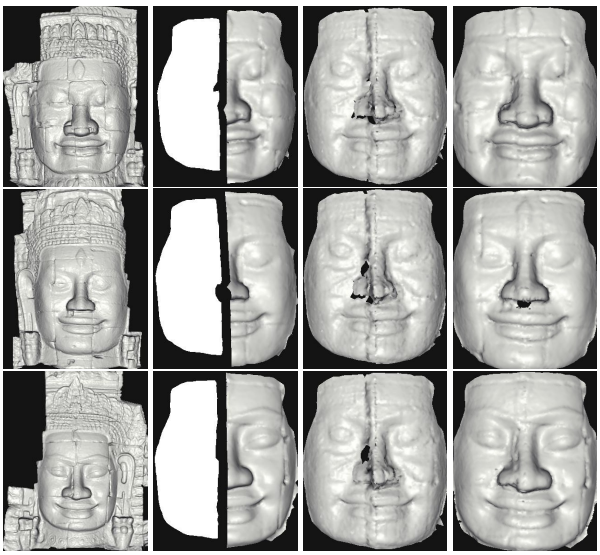


Fig. 7: An experiment with known ground truth to verify the correctness of the shape restoration method proposed in Section 3.3.2. Left: original shapes. Middle left: synthesized inputs, with nearly half of each face cropped. The missing part of each face corresponds to a plane on the surface of its bounding box. Middle right: corresponding restored outputs of the original matrix recovery method (Section 3.3.1). Right: restoration results of the restoration method we proposed.

We would like to examine the effectiveness of our proposed method in Section 3.3.2. A test with the Bayon database was designed as below. First, we chose several samples to form a small data set. Six well-preserved faces in total were chosen from the entire database. For each chosen face, five extra samples were synthesized by cutting down a certain area randomly, as illustrated in Fig. 6. Holes were filled using simple smooth surfaces. This results in a data set containing 36 samples, on which both the original matrix recovery method in Section 3.3.1 and our proposed method in Section 3.3.2 were carried out, aiming to restore those incomplete samples back to their origins.

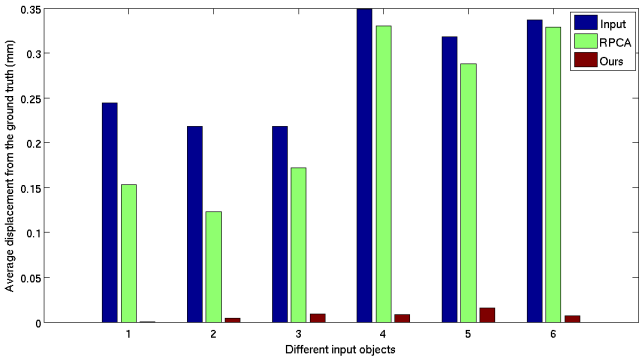
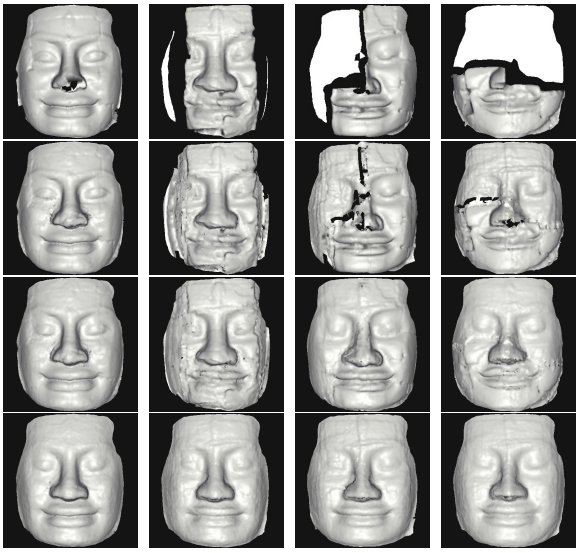


Fig. 8: Comparison of the restroration accuracy. For each individual sample, the average displacements between different restoration results and the ground truth are shown. According to this result, our restoration algorithm beats the original robust PCA method in all these cases.

The process of restoring this small data set cost less than 20 seconds on a common laptop with our *Matlab* implementation. Part of the restored outputs are shown in Fig. 7. Compared with the outputs of the original matrix recovery model, which look quite similar to each other, more accurate results were achieved by our revised algorithm introduced in Section 3.3.2. A comparison of the restroration accuracy is shown in Fig. 8. This verifies the effectiveness of the proposed shape restoration method.

3.4.2 Restore the Bayon Faces

More restoration results were generated using the entire Bayon face database. In the convex optimization process (Eq. (15)), there is a weighting parameter λ that trades off the rank of the solution versus the sparseness of the error. As we mentioned, parameter λ is in the form $c/\sqrt{m}$, where c is a constant, typically set to 1. m is the length of the input vectors, fixed to three times the number of corresponding points in our experiment. Therefore the constant c could be used as a scaled version of the parameter λ. Notice that for our shape restoration problem, this parameter c trades off the similarities of all input models versus the characteristics of each sample: the larger c is, the more individual characteristics, as well as the error caused by shape incompletion, will be kept and vice versa. Fig. 9 illustrates the effect of changing the value of parameter c. The result shows that the typical value 1 seems to be a good trade-off for parameter c in our shape restoration task.



(a) *No.* 20E (b) *No.* 22E (c) *No.* 22N (d) *No.* 24E

Fig. 9: More restoration results, with three different values of parameter c. Each column belongs to the same sample, and the first row shows the original inputs. The remaining rows demonstrate the outputs under different values of parameter c, 2, 1.6 and 1, respectively, from top to bottom.

4. Joint Restoration with Cluster Analysis

In the previous section, we proposed a novel method that builds upon the extended matrix recovery theory to restore incomplete samples. However, if there are two or more subcategories in the given data set, it is more reasonable to take only the samples within that certain subcategory into consideration, instead of using the entire data set, since these objects from the same subcategory are more close to each other than the rest. Moreover, sometimes we only know that there are some different subcategories among the given data set, without the accurate grouping information, since some samples may be partially damaged! This makes the matter even worse, and it is quite common when dealing with culture relics.

In this section, we introduce a novel comprehensive method to solve this kind of problem, combining the previous restoration method and cluster analysis together. Accurate restoration is achieved by adopting a two-step shape recovery strategy. Rough restoration and clustering processes are first carried out using the entire database to group similar samples together. Then refined restoration using high resolution data is executed in each cluster to restore higher details while retaining the characteristics of each shape.

4.1 Restoration and Clustering

As we mentioned before, in order to achieve detailed restoration results, the given data set should be divided according to the shapes of samples. This is a typical task of cluster analysis, with several classical solutions, such as *K-means* discussed in [5]. Here we choose a flexible clustering method—*hierarchical clustering*. Instead of generating a "flat" data description, like K-means, hierarchical clustering lead to a hierarchical representa-

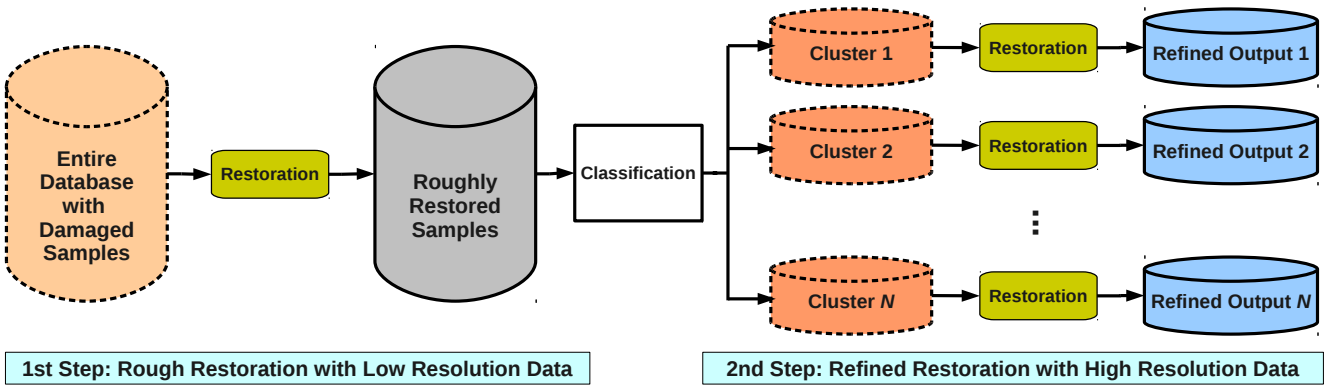


Fig. 10: The workflow of our proposed two-step restoration method. The same restoration algorithm, which is explained in Section 3.3.2, is used in both steps.

tion, where the relationship among all samples is clear at a glance and clusters can be chosen more flexibly.

In the practice of refined restoration, an additional benefit is that for a certain sample, which is one leaf node in the dendrogram, the scope of the cluster used for recovery can be gradually enlarged, until a satisfactory result appears. This bottom-up strategy also enables efficient computation on limited memory, as only a part of the database is required in the computation.

We choose a simple but effective approach to evaluate similarity between two samples. After reforming each sample as a length-fixed vector, the similarity evaluation turns into the problem of measuring distance between two points in a high-dimensional vector space. Here the Euclidean distance is employed among a number of possible metric candidates. Notice that the raw 3D measurements should not be used to similarity evaluation directly, since some of them may be incomplete. Instead, a rough shape restoration procedure is carried out beforehand, using all but relatively low resolution data.

4.1.1 Integrating Restoration with Clustering

We leverage the fact that similar shapes can be restored via the method introduced in the previous section. Considering that these samples may come from different subcategories, a two-step restoration strategy is developed, as shown in Fig. 10.

First of all, a preprocessing similar as Section 3.2 is adopted, turning each sample into a fixed-length vector to represent its geometric shape. After that, in order to properly divide the data set into subcategories, a rough restoration process with all samples is carried out to temporarily handle data incompletion. Then a refined restoration with higher resolution data is executed in each respective cluster, which results in detailed restorations that retain the characteristics of each face. Several advantages are brought by the proposed method, including higher restoration accuracy and reduced computation. Notice that we adopt the same restoration method in both rough and refined shape recovery steps using data with different resolution, and the scopes of the faces within the database are different, too.

4.1.2 Post-processing

Recall that in the optimization Eq. (15), the output matrix E evaluates the difference between observation D and estimated original shape A, and Ω^c records the positions of missing entries. This can be used to help doing the merging process. In details, approximate confidence scores can be estimated for sampling points on both restored and initial shapes. Those points with higher confidence scores are supposed to dominate the merging process. It is clear that for restored points corresponding to missing data, their scores of confidence should be set higher. Similarly, for a certain observed entry (i, j) within Ω, if the difference value e_{ij} is relatively large, showing that there might be a corruption, the confidence value of this point from the restored shape should be set higher as well.

4.2 Experiments

In this section, we show the results of applying the proposed method to the Bayon face database. 161 digital copies of scanned faces are included in this data set. Due to the cropping operation in the preprocessing step, our restoration method handles the facial part only, without considering outer parts such as ears and headdresses.

4.2.1 Experiments with Bayon Faces

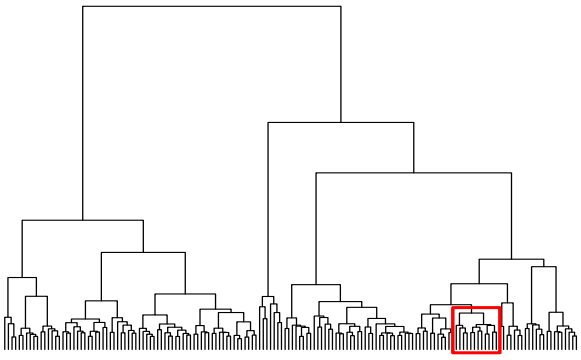


Fig. 11: The hierarchy of Bayon faces, based on the global rough restoration results using low resolution data.

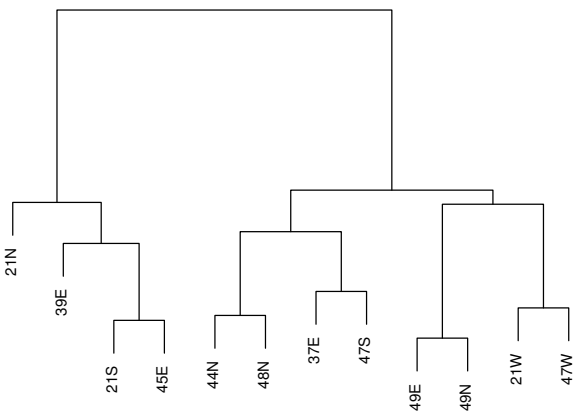


Fig. 12: Enlarged image of the hierarchy corresponding to the chosen cluster in Fig. 11.

First, after the rough restoration and classification procedures, similar samples were grouped together to generate a hierarchy shown in Fig. 11. Here we randomly chose one cluster in the whole dendrogram, which is marked by a red box in that figure, to demonstrate the restored results of our method. The enlarged image of the hierarchy corresponding to this cluster is shown in Fig. 12. Notice that due to the characteristic of our restoration method, a necessarily large cluster, which usually contains more than three samples, is required. Moreover, if the restoration result is not satisfactory enough (judged by the user), the cluster can be gradually enlarged until good results are generated.

Fig. 13: The 12 samples that belong to the chosen cluster in Fig. 11.

There are 12 samples in the chosen cluster, where several samples are partially damaged, as shown in Fig. 13. For example, the third face in the bottom row, which is numbered as No. 49E, has a broken nose. After applying the refined restoration process to this group, the missing nose part of this sample is recovered, as shown in Fig. 14. From the comparison from both frontal and side views, as well as the enlarged partial details, we can see that

Fig. 14: One experimental result of our proposed restoration method, face No. 49E, with comparison from a side view and enlarged partial details. Upper: measured shape. Bottom: restored output.

the broken nose and mouth are well restored, without changing other undamaged areas.

4.2.2 Clustering Bayon Tower

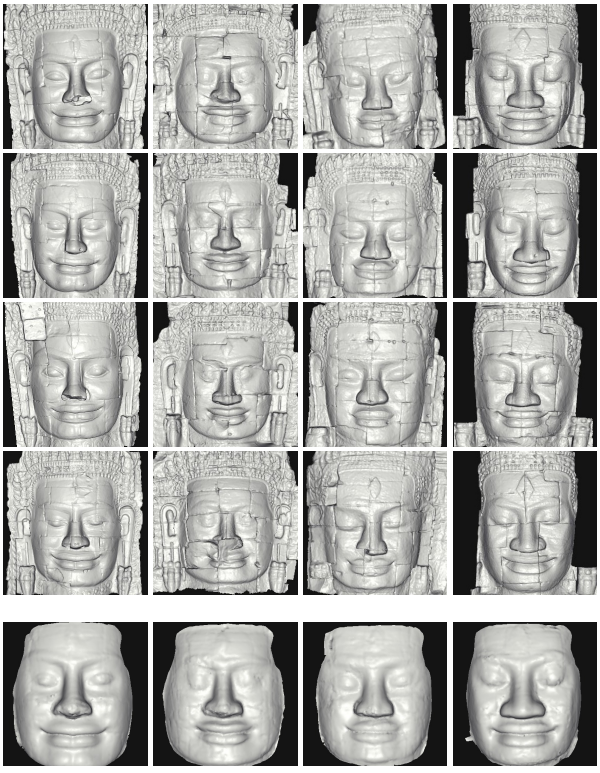


Fig. 15: Similarities within towers. Each column corresponds to a certain face tower: from top to bottom, the first four items show the four facial sculptures on that tower, and the last one illustrates the tower's representative shape (average shape).

We also applied our method to cluster Bayon face towers. According to [27], facial sculptures on the same tower are of a significant similarity comparing to those from different ones. Based on this fact, we presented a hypothesis where each face tower

could be treated as an unit, and further modified the original goal to classifying face towers. In order to eliminate the impact of data incompletion, our restoration scheme was adopted within each tower. Towers were represented by the average shapes of restored faces, and then hierarchically clustered based on their pairwise similarities. The spatial distribution of this classification result shows that there are some patterns among the structure of Bayon that could be found to be meaningful through further archaeological research.

The famous face towers are located on the upper terrace of the Bayon. Typically, each tower supports four huge smiling faces. In the previous classification studies, such as [35] and [27], faces were treated individually without considering the spatial connections. However, it is easy to notice that faces on the same tower look more similar than those from different ones, as shown in Fig. 15. Hence we may assume that, compared with considering every face separately, it is more reasonable to treat each face tower as a union, as it seems to have a high shape similarity within each tower.

There are at least two advantages to consider each tower as a unit: first, the number of clustering units is reduced, making the result clear at a glance, and it is therefore easy to discover the hidden regularities; second, a number of faces are only partially preserved, which makes the comparison inaccurate and inappropriate. If every tower is treated as a whole, then the other complete faces on the same tower can be used to help repairing the missing part, following the restoration scheme introduced in Section 3.3.2.

A restoration procedure introduced in Section 3.3.2 was adopted within each face tower, trying to handle the shape incompletion and make the similarity comparison more accurate.

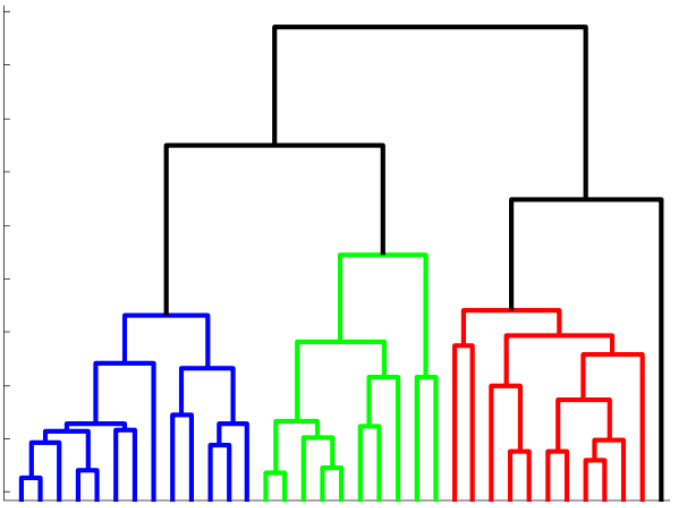


Fig. 16: The hierarchy of Bayon face tower clusters. This dendrogram was divided into three groups, labeled in different colors. The numbers on the bottom axis refer to the serial numbers of face towers; while the vertical axis refers to the distance measure between spots or spot clusters. The height of a node can be thought as the distance value between its right and left sub-branch clusters. Notice that tower *No. 52* was labeled as an outlier in this result.

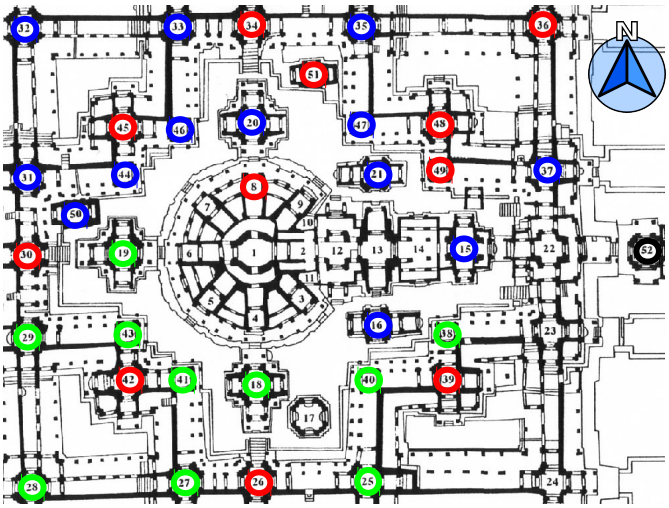


Fig. 17: The distribution of our clustering result showed in Fig. 11 on the plane of Bayon.

The average of repaired faces was used for representing the corresponding face tower. In case no other than one face existed in a certain tower (which is really rare in our experiment), the restoration step would be skipped, keeping the original shape for further comparison.

After the proceedings described in Section 3.2, a small data set containing the representative faces of 35 towers was built. Based on this data set, a hierarchy of face tower clusters was generated (Fig. 16). The tree root refers to a single cluster containing all elements, and the leaves correspond to individual face towers. After setting a proper threshold value, the whole dendrogram was divided into three groups, labeled in different colors. Fig. 17 illustrates the distribution of clustering results on the plane figure of Bayon. Notice that tower *No. 52* was included in none of the above three groups, because it was "far" from all the other towers, which means it had a notable difference from the rest. This was also confirmed by JSA experts. From their perspective, this tower seemed to be built later than the others, with a slightly worse sculpting skill compared with all the other towers.

After acquiring the above classification results, we tried to unscramble them from the perspective of culture and history, hoping to get some new findings of the ancient mystery.

First of all, our classification results were consistent with the three-type hypothesis of Bayon faces given by the JSA report [35]. That was that these faces can be roughly divided into three basic groups, as shown in Fig. 11. This indicates that there might be different sculpture teams building the Bayon at the same time.

Second, it is interesting to find that the spatial distribution of our result shows a certain degree of structural information. Let us focus on Fig. 17, it seems that the blue and green groups are almost symmetrically distributed on both sides of the east-west axis of this temple. Moreover, for the third group (the red one), all samples, except towers *No.* 8, 36, and 48, are located on the symmetrical positions, forming a figure that resembles the meaningful structure called a "Mandala".As this kind of structure is widely used in the Hindu and Buddhist religious traditions for establishing a sacred space, the above discovery confirmed our classification results to some extent.

5. Intraclass Shape Comparison

In the previous two sections, we solved the problem about how to remove corruptions from a given data set using the hidden group similarities. Complete shapes can be recovered even when there are two or more unknown subcategories exist. Now we move on to the further analysis: to compare similar shapes quantitatively. Compared with general shape comparison problems, intraclass shape analysis enjoys better semantic explanation and higher processing accuracy, which bring it increasing concern with wide range of applications, such as biometrics, medical diagnosis, manufacturing industry, etc.

Given a group of shapes for comparison, denoted as $\{S_i\}_{i=1}^N$, the traditional method is to find a proper metric $d : S \times S \rightarrow \mathbf{R}$ to evaluate the distance between them. This metric can be used to form a distance matrix, like the one shown in Fig.18.

However, if we want to know the details about shape differences, the previous scalar evaluation will be no longer enough. For instance, given several silhouettes from different kinds of butterfly, as shown in Fig.19, we would like to detect the most identifiable regions that cause the subcategories differ from each other. In this case, the feature region of the *Papilio* samples is the *swallowtail* highlighted in the figure.

5.1 Shape Matching

One essential part of the proposed comparison method is shape registration. While source object is deformed to target shape, the differences between them will be revealed.

Here we choose a non-rigid registration method introduced in [21]. Comparing with other shape matching method, this method is more appropriate for our problem. This is mainly because that common methods usually focus on reducing the residuals of registration only, without considering of keeping local structures. This may lead to inaccurate correspondences between objects and hurt the performance of the later comparison. On the contrary, based on the observation that similar objects usually have very similar local structures, the chosen registration method uses local rigid transformations to guide a underlying free-form deformation (FFD, introduced in [42]). Due to this attribute, local structures will be kept during the registration.

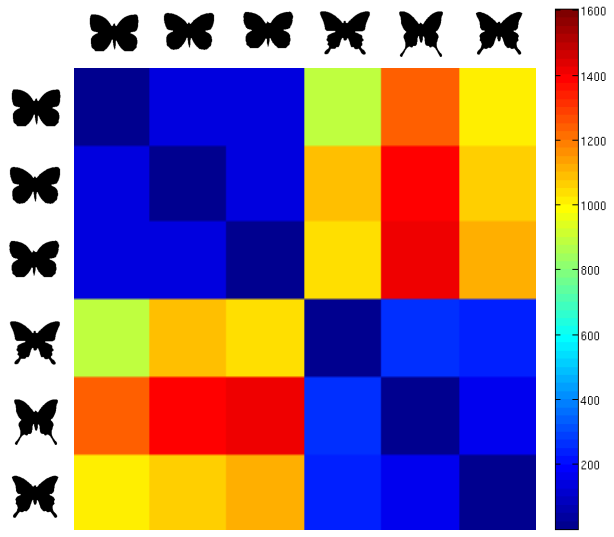


Fig. 18: An example of distance matrix

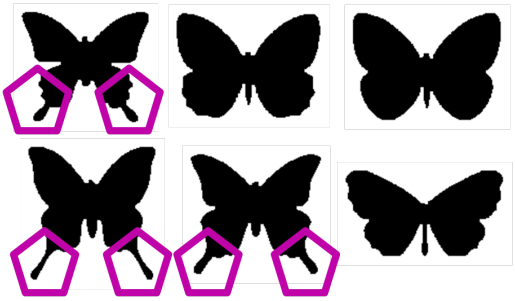


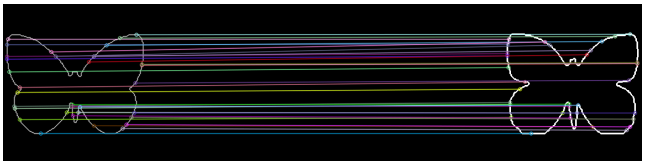
Fig. 19: Silhouettes of butterfly. The highlighted swallowtail regions make the *Papilio* samples different from the others.

Moreover, the registration process also provides us a way to find correspondences between similar objects. Based on the registered result, corresponding points can be obtained by adopting nearest neighbor point search. Fig.20 shows an example of shape matching.

Notice that in the above shape matching method, a certain sample is chosen as the template, to which the rest all samples will be

(a) template (b) target

(c) contours before registration (d) contours after registration



(e) matching result

Fig. 20: An example of shape matching. The correspondences are obtained via a nearest neighbor search based on the registered shape (d). A common method to accelerate this search is to use k-d tree data structure.

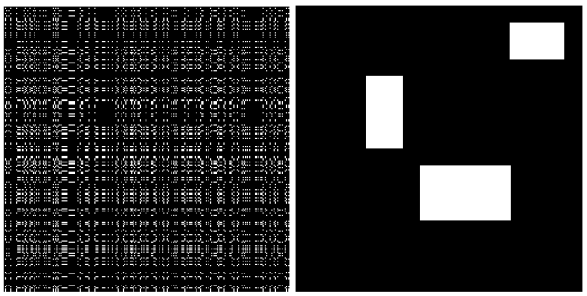
registered. This will actually affect the shape matching process, since different templates will lead to different corresponding results. Fortunately, the difference is small enough to tolerant for our analysis.

5.2 Identifiable Region Detection

Traditional cluster analysis can only provide overall grouping information, such as the dendrogram generated by hierarchical clustering. In this section, we will introduce a method to automatically detect identifiable regions among a group of shapes.

5.2.1 Biclustering Methods

Recently, biclustering techniques, such as studies in [14], [20], [33], [39], were proposed for revealing submatrices showing unique patterns, e.g. a submatrix with low numerical rank [44]. Different from traditional clustering method, these methods simultaneously discover row and column groups and the detected biclusters may correspond to arbitrary subsets of rows and columns, as the white rectangles shown in Fig.21.



(a) Original matrix (b) Biclustering result

Fig. 21: A demonstration of the biclustering process. After proper operations to re-order both rows and columns of the original matrix, a biclustering result is obtained with clearer meaning and structures.

In our problem, we choose the biclustering method proposed in [14]. Based on the fact that good clustering usually leads to a few homogeneous blocks in data matrix, this method makes use of lossless data compression to decompose a binary matrix into disjoint row and column groups. Comparing with other biclustering method, numbers of row and column groups are no longer needed to be specified in this method.

5.2.2 Detecting Identifiable Region via Biclustering

Based on the correspondences we obtained by shape matching, we start to find feature regions that distinguish different subcategories from each other.

Suppose there are N targets, each of which has M sampling points with known correspondences. We first generate a N-by-M matrix D to record detailed differences when these targets are compared to a certain object, e.g. the average shape. In order to emphasize the most notable regions, we use the robust PCA method introduced in [12] to filter this matrix, highlighting significant regions.

We then search for notable similarities along not only the dimension of the object index but also the dimension that correspond to spatial sampling points. Due to the fact that identifiable regions lead to regular structures in the difference matrix we generated, we turn this partial similarity searching problem into a low-rank submatrix detection process, which can be solved by biclustering method. Algorithm 3 shows the whole flow of our proposed method.

Algorithm 3 Identifiable Region Detection

Input: a group of similar objects $\{S_i\}_{i=1}^N$

Obtain dense correspondences $\{P_{S_i}\}_{i=1}^N$ between $\{S_i\}_{i=1}^N$

Calculate the average $\overline{P_S} = \frac{1}{N}\sum_{i=1}^N P_{S_i}$

for $i = 1 \rightarrow N$ **do**

Evaluate the displacements at every corresponding points:

$\mathbf{d}_{S_i} = |P_{S_i} - P_S|$

end for

Form the displacement matrix $D \doteq [\mathbf{d}_{S_1}|\cdots|\mathbf{d}_{S_N}]$

Filter D to matrix A in order to highlight features

Binarize A to matrix B by thresholding

Adopting biclustering on B to find notable regions $\{R_t\}_{t=1}^K$

Output: $\{R_t\}_{t=1}^K$

5.3 Experiments

Here comes the experiment part. From a large binary image database collected by the LEMS Vision Group at Brown University, we selected a butterfly silhouette data set, as shown in Fig.22. Samples from several subcategories are included, such as Papilio with swallowtail. The following experiments are mainly adopted on this data set.

We adopted the proposed method to the butterfly data set. Fig.23 show the effectiveness of the low-rank matrix filtering we designed. Two low-rank submatrices are found in the biclustering process, as shown in Fig.24, which correspond to two different subcategories respectively. Notice that the desired result similar to Fig.19 is obtained.

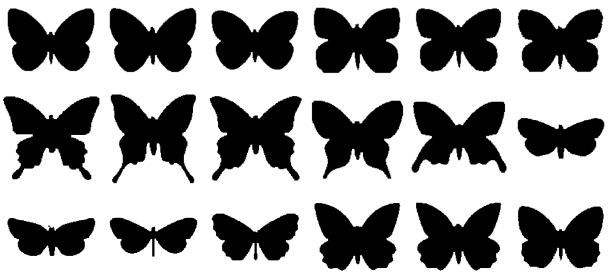
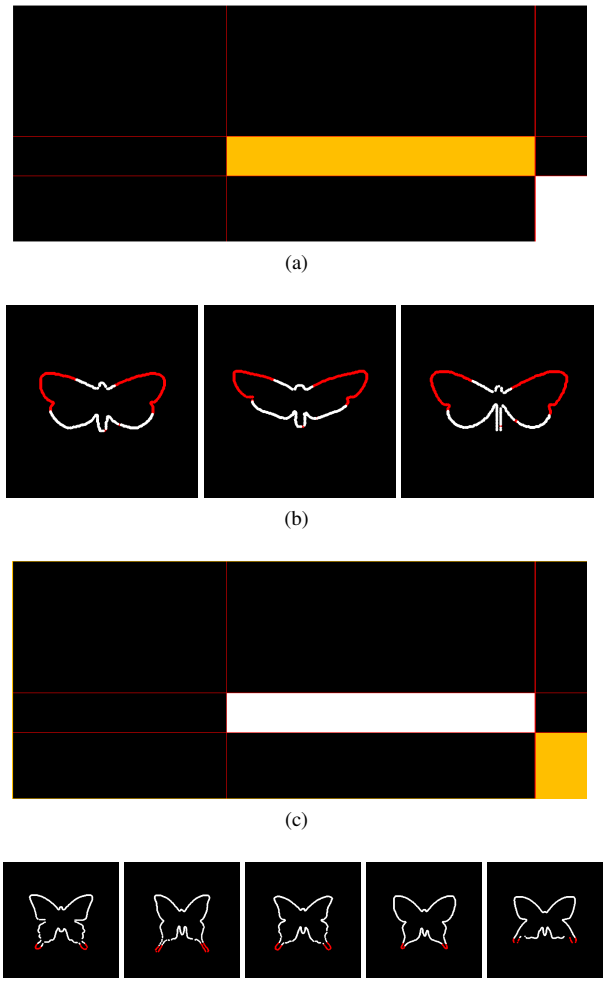


Fig. 22: A butterfly data set from a large silhouette image database collected by the LEMS Vision Group at Brown University.

(a) Original binary representation matrix

(b) Representation matrix after RPCA filtering

Fig. 23: The effect of RPCA filtering. Clearer result is generated.



(a) (b) (c) (d)

Fig. 24: The two detected low-rank submatrices (yellow blocks in the figure) and their corresponding regions (highlighted in red color).

6. Conclusions

In this paper, we focus on the general problem of intraclass shape analysis. Given samples from an arbitrary category, we would like to discover the overall similarity and the detailed description about shape difference. In case that defective samples exist, we handle shape incompletion as well. Besides, we can further divide the data set into sub-categories according to their similarities.

More specifically, we adopt a non-rigid matching scheme to align sample shapes. As for the shape restoration issue, unlike existing example-based approaches, our method simultaneously restores incomplete samples via convex optimization, without manual intervention and prior knowledge about shape. We also show how to combine restoration with cluster analysis, which further improve the restoration. Moreover, based on the deformation field from pairwise shape comparison, we form a similarity tensor to record shape differences. Low-dimensional shape representation can be obtained via tensor decomposition. Besides, we turn the identifiable region detection problem into a low-rank submatrices searching process, and solve it with biclustering algorithm. The proposed methods are evaluated with real world data. Satisfactory results are achieved, which verifies the effectiveness of our study.

6.1 Limitations

Although the effectiveness of the proposed methods is proved by experimental results, there are still several limitations:

Poor restoration outputs may occur

Due to the fact that our method makes use of group similarity implicitly, a necessarily large data set containing enough samples, usually more than three, is required. For example, if a certain part of all samples are missing, it will be really difficult for our method to recover that area, since there is no information about that missing part.

Restored objects are smoothed to some extent

Currently our method is good at keeping common attributes among a group of similar objects, while the individual characters are difficult to extract. This will lead to smoothed restoration, with some details lost. Besides, the current shape matching method relies on the template chosen, which should be refined as well.

6.2 Extensions and Open Problem

We hope that this paper demonstrates the utility of low-rank matrix recovery theory in the shape analysis domain. Here we describe several possible extensions to our work, as well as some possible directions for future research.

To integrate shape matching into optimization

Since the proposed restoration method requires accurate shape correspondences, while currently we only do the shape matching once, before the optimization. If we could integrate the non-rigid shape matching into the whole

optimization and keep updating the correspondence, the restoration result would be more accurate.

Iteratively restoring and clustering

For the restoration and clustering problem discussed in Section 4, our proposed joint method works not iteratively, which is more effective for this kind of chicken-and-egg problem. This can be improved in future.

To further develop shape comparison method

As for the identifiable region detection method discussed in Section 5, spatial constraints should be added during the biclustering process. This actually can be solved by adopting a shape segmentation first, and then cluster these pieces.

Acknowledgments We would like to acknowledge our excellent collaborators, Kent Fujiwara and Mawo Kamakura, for their hard work and useful feedback to this research. This work is partially supported by the *Strategic Information and Communications R&D Promotion Programme* (SCOPE).

References

[1] Alvaro, C., Claudio, E., Antonio, O. and Paulo, R. C.: 3D As-Rigid-As-Possible Deformations Using MLS, *The 25th Computer Graphics International Conference (CGI'2007)* (2007).
[2] Belongie, S., Malik, J. and Puzicha, J.: Shape Matching and Object Recognition Using Shape Contexts, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 24, No. 4, pp. 509–522 (2002).
[3] Berretti, S., Del Bimbo, A. and Pala, P.: Retrieval by shape similarity with perceptual distance and effective indexing, *Trans. Multi.*, Vol. 2, No. 4, pp. 225–239 (2000).
[4] Besl, P. J. and McKay, N. D.: A method for registration of 3-D shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, pp. 239–256 (1992).
[5] Bishop, C. M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006).
[6] Boyd, S. and Vandenberghe, L.: *Convex Optimization*, Cambridge University Press, New York, NY, USA (2004).
[7] Breckon, T. P. and Fisher, R. B.: Three-Dimensional Surface Relief Completion Via Nonparametric Techniques, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, pp. 2249–2255 (2008).
[8] Bronstein, A. M. and Bronstein, M. M.: Regularized Partial Matching of Rigid Shapes, *ECCV (2)* (Forsyth, D. A., Torr, P. H. S. and Zisserman, A., eds.), Lecture Notes in Computer Science, Vol. 5303, Springer, pp. 143–154 (2008).
[9] Bronstein, A. M., Bronstein, M. M., Bruckstein, A. M. and Kimmel, R.: Analysis of Two-Dimensional Non-Rigid Shapes, *International Journal of Computer Vision*, Vol. 78, No. 1, pp. 67–88 (2008).
[10] Bronstein, A. M., Bronstein, M. M., Bruckstein, A. M. and Kimmel, R.: Partial Similarity of Objects, or How to Compare a Centaur to a Horse, *Int. J. Comput. Vision*, Vol. 84, No. 2, pp. 163–183 (2009).
[11] Bronstein, A. M., Bronstein, M. M., Guibas, L. J. and Ovsjanikov, M.: Shape google: Geometric words and expressions for invariant shape retrieval, *ACM Trans. Graph.*, Vol. 30, No. 1, pp. 1:1–1:20 (2011).
[12] Candès, E. J., Li, X., Ma, Y. and Wright, J.: Robust principal component analysis?, *Journal of the ACM*, Vol. 58, pp. 11:1–11:37 (2011).
[13] Candes, E. J. and Recht, B.: Exact Matrix Completion via Convex Optimization, *Found. Comput. Math.*, Vol. 9, No. 6, pp. 717–772 (2009).
[14] Chakrabarti, D., Papadimitriou, S., Modha, D. S. and Faloutsos, C.: Fully automatic cross-associations, *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, ACM, pp. 79–88 (2004).
[15] Cunin, O.: The Bayon: an Archaeological and Architectural Study, *Bayon: New Perspectives*, River Books, pp. 136 – 229 (2007).
[16] Davis, J., Marschner, S., Garr, M. and Levoy, M.: Filling holes in complex surfaces using volumetric diffusion, *The First International Symposium on 3D Data Processing Visualization and Transmission*, pp. 428 – 861 (2002).
[17] Dijkstra, E. W.: A note on two problems in connexion with graphs, *Numerische Mathematik*, Vol. 1, pp. 269–271 (1959).
[18] Dryden, I. L. and Mardia, K. V.: *Statistical Shape Analysis*, John Wiley & Sons (1998).
[19] Elhamifar, E. and Vidal, R.: Block-Sparse Recovery via Convex Optimization, *IEEE Transactions on Signal Processing*, Vol. 60, No. 8, pp. 4094–4107 (2012).
[20] Eriksson, B., Balzano, L. and Nowak, R.: High-Rank Matrix Completion, *Journal of Machine Learning Research - Proceedings Track*, Vol. 22, pp. 373–381 (2012).
[21] Fujiwara, K.: Non-rigid registration for shape analysis, PhD Thesis, The University of Tokyo (2013).
[22] Funkhouser, T. and Shilane, P.: Partial matching of 3D shapes with priority-driven search, *Proceedings of the fourth Eurographics symposium on Geometry processing*, pp. 131–142 (2006).
[23] Gower, J.: Generalized procrustes analysis, *Psychometrika*, Vol. 40, No. 1, pp. 33–51 (1975).
[24] Johnson, A. E. and Hebert, M.: Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 5, pp. 433–449 (1999).
[25] Jolliffe, I.: *Principal Component Analysis*, Springer-Verlag, second edition edition (2002).
[26] Ju, T.: Robust repair of polygonal models, *ACM Transactions on Graphics*, Vol. 23, No. 3, pp. 888–895 (2004).
[27] Kamakura, M., Oishi, T., Takamatsu, J. and Ikeuchi, K.: Classification of Bayon Faces Using 3D Models, *The 11th International Conference on Virtual Systems and Multimedia (VSMM' 05)* (2005).
[28] Klare, B., Mallapragada, P., Jain, A. K. and Davis, K.: Clustering face carvings: Exploring the devatas of Angkor Wat, *Proceedings of the 20th International Conference on Pattern Recognition*, Washington, DC, USA, IEEE Computer Society, pp. 1517–1520 (2010).
[29] Kraevoy, V. and Sheffer, A.: Template-based mesh completion, *Proceedings of the third Eurographics symposium on Geometry processing (SGP' 05)* (2005).
[30] Liepa, P.: Filling holes in meshes, *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing (SGP' 03)*, pp. 200–205 (2003).
[31] Lin, Z., Chen, M. and Ma, Y.: The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices, Technical Report UILU-ENG-09-2215, University of Illinois at Urbana-Champaign (2009).
[32] Lipman, Y. and Funkhouser, T.: Möbius voting for surface correspondence, *ACM SIGGRAPH 2009* (2009).
[33] Mackey, L. W., Talwalkar, A. and Jordan, M. I.: Divide-and-Conquer Matrix Factorization, *NIPS* (Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F. C. N. and Weinberger, K. Q., eds.), pp. 1134–1142 (2011).
[34] Mancera, L. and Portilla, J.: L0-Norm-Based Sparse Representation Through Alternate Projections, *ICIP*, IEEE, pp. 2089–2092 (2006).
[35] Nakagawa, T.(ed.): *Annual report on the technical survey of Angkor Monument*, Japan International Cooperation Center (1998).
[36] Nooruddin, F. S. and Turk, G.: Simplification and Repair of Polygonal Models Using Volumetric Techniques, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 9, No. 2, pp. 191–205 (2003).
[37] Pauly, M., Mitra, N. J., Giesen, J., Gross, M. and Guibas, L. J.: Example-based 3D scan completion, *Proceedings of the third Eurographics symposium on Geometry processing (SGP' 05)* (2005).
[38] Pekelny, Y. and Gotsman, C.: Articulated Object Reconstruction and Markerless Motion Capture from Depth Video, *Comput. Graph. Forum*, Vol. 27, No. 2, pp. 399–408 (2008).
[39] Rangan, A. V.: Short note: A simple filter for detecting low-rank submatrices, *J. Comput. Phys.*, Vol. 231, No. 7, pp. 2682–2690 (2012).
[40] Reuter, M.: Hierarchical Shape Segmentation and Registration via Topological Features of Laplace-Beltrami Eigenfunctions, *Int. J. Comput. Vision*, Vol. 89, No. 2-3, pp. 287–308 (2010).
[41] Schaefer, S., McPhail, T. and Warren, J.: Image deformation using moving least squares, *ACM Trans. Graph.*, Vol. 25, No. 3, pp. 533–540 (2006).
[42] Sederberg, T. W. and Parry, S. R.: Free-form deformation of solid geometric models, *SIGGRAPH Comput. Graph.*, Vol. 20, No. 4, pp. 151–160 (1986).
[43] Sharf, A., Alexa, M. and Cohen-Or, D.: Context-based surface completion, *ACM Transactions on Graphics*, Vol. 23, pp. 878–887 (2004).
[44] Strang, G. and Nguyen, T.: The Interplay of Ranks of Submatrices, *SIAM Review*, Vol. 46, No. 4, pp. 637–646 (2004).
[45] Uchida, E., Ogawa, Y., Maeda, N. and Nakagawa, T.: Deterioration of stone materials in the Angkor Monuments, Cambodia, *Engineering Geological Advances in Japan for the New Millennium*, Elsevier, pp. 329 – 340 (2000).

[46] van Kaick, O., Zhang, H., Hamarneh, G. and Cohen-Or, D.: A Survey on Shape Correspondence, *Comput. Graph. Forum*, Vol. 30, No. 6, pp. 1681–1707 (2011).
[47] Wang, Y., Liu, J. and Tang, X.: Robust 3D Face Recognition by Local Shape Difference Boosting, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 32, No. 10, pp. 1858–1870 (2010).
[48] Wright, J., Ganesh, A., Rao, S., Peng, Y. and Ma, Y.: Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization, *Advances in Neural Information Processing Systems 22*, MIT Press (2009).
[49] Wright, J. and Ma, Y.: Dense error correction via l^1-minimization, *IEEE Transactions on Information Theory*, Vol. 56, No. 7, pp. 3540–3560 (2010).
[50] Xu, K., Zhang, H., Tagliasacchi, A., Liu, L., Li, G., Meng, M. and Xiong, Y.: Partial intrinsic reflectional symmetry of 3D shapes, *ACM Trans. Graph.*, Vol. 28, No. 5, pp. 138:1–138:10 (2009).
[51] Zeng, W., Zeng, Y., Wang, Y., Yin, X., Gu, X. and Samaras, D.: 3D Non-rigid Surface Matching and Registration Based on Holomorphic Differentials, *ECCV '08: Proceedings of the 10th European Conference on Computer Vision* (2008).
[52] Zeng, Y., Gu, X., Samaras, D., Wang, C., Wang, Y. and Paragios, N.: Dense Non-rigid Surface Registration Using High-Order Graph Matching, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (2010).
[53] Zhang, H., Sheffer, A., Cohen-Or, D., Zhou, Q., van Kaick, O. and Tagliasacchi, A.: Deformation-Driven Shape Correspondence, *Computer Graphics Forum (Special Issue of Symposium on Geometry Processing 2008)*, Vol. 27, No. 5, pp. 1431–1439 (2008).