Putting the Pieces Together: Regularized Multi-part Shape Matching

Or Litany[1], Alexander M. Bronstein[1], and Michael M. Bronstein[2]

[1] School of Electrical Engineering, Tel Aviv University, Israel
[2] Institute of Computational Science, Faculty of Informatics
Universita della Svizzera Italiana, Lugano, Switzerland
Paper ID 314

Abstract. Multi-part shape matching is an important class of problems, arising in many fields such as computational archaeology, biology, geometry processing, computer graphics and vision. In this paper, we address the problem of simultaneous matching and segmentation of multiple shapes. We assume to be given a reference shape and multiple parts partially matching the reference. Each of these parts can have additional clutter, have overlap with other parts, or there might be missing parts. We show experimental results of efficient and accurate assembly of fractured synthetic and real objects.

1 Introduction

Multi-part shape matching is an important class of problems, arising in computational archaeology (assembly of fractured objects [9,19,18,10]), computational biology (protein docking [12]), and computer vision (merging of partial 3D scans [17,21] and assembling 2D and 3D puzzles [13,7]). Traditionally, the matching of rigid 3D shapes has been performed using variants of the classical *iterative closest point* (ICP) algorithm [2,6,15,8], trying to optimally align the shapes by means of

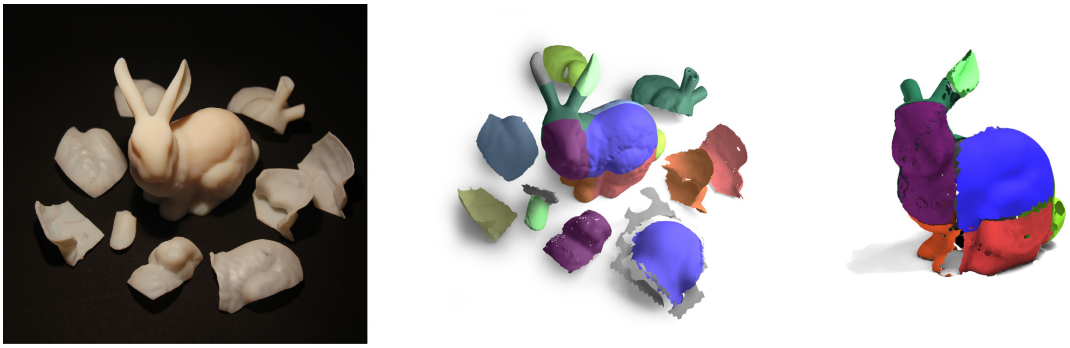

Fig. 1. Assembling the Stanford bunny: 3D-printed bunny fractured into multiple parts (left); each part is scanned individually with clutter (center); the parts are matched to the reference shape (right) using the proposed approach

A. Fusiello et al. (Eds.): ECCV 2012 Ws/Demos, Part I, LNCS 7583, pp. 1–11, 2012.

a rigid transformation minimizing a surface-to-surface distance between them. Matching of multiple shapes has been done using graph-based methods [11], iterative pairwise part registration [21], as well as "multi-part ICP" where optimization is performed over rigid transformation parameters of all parts [14]. In [20], a correspondence-less partial matching was proposed using optimization over parts that maximize the similarity of local descriptors.

In the partial matching setting when parts are missing (e.g. due to occlusions in the scan process) or conversely, clutter is present (e.g. if each part is scanned individually), the ICP algorithms can be modified by introducing weights that reject points with a "bad" correspondence. The shortcoming of such weighting is that it does not allow direct control of the size and regularity of the matching parts. A remedy to this problem was proposed in [5], where the authors used the partial matching framework of [3], optimizing simultaneously for part similarity, size, and regularity.

Here, we extend this approach to multiple parts, performing matching and segmentation of multiple shapes at the same time. In the setting we address, we are given a reference shape and multiple parts partially matching the reference. Each of these parts can have additional clutter, have overlap with other parts (like in the 3D view merging); furthermore, there might be uncovered parts of the reference shape. Some of the applications of the presented method arise, for example, in assisting orthopedic surgeons in putting fragmented bones back together using a healthy bone 3D model (taken pre-trauma or from a symmetric bone). Another uses are in automated car parts assembly.

2 Background

Given two rigid shapes X and Y, a standard way to match them is to look for a rigid transformation $\mathbf{T}$ (rotation+translation) of one of them (w.l.o.g. Y) such that some distance between $\mathbf{T}Y$ and X is minimized,

$$\min_{\mathbf{T}\in \mathrm{SE}(3)} D(X, \mathbf{T}Y). \tag{1}$$

The shape-to-shape distance D can be e.g. the *Hausdorff distance*,

$$D_{\mathrm{H}}(X,Y) = \max\left\{ \max_{x\in X}\min_{y\in Y}\|x-y\|, \max_{y\in Y}\min_{x\in X}\|x-y\| \right\}.$$

Denoting by $y^*(x) = \mathrm{argmin}_{y\in Y}\|x-y\|^2$ and $x^*(y) = \mathrm{argmin}_{x\in X}\|x-y\|^2$ the closest point from x on Y and from y on X, respectively, D_{H} can be written as

$$D_{\mathrm{H}}(X,Y) = \max\left\{ \max_{x\in X}\|x-y^*(x)\|, \max_{y\in Y}\|x^*(y)-y\| \right\}.$$

This formulation allows solving (2) by means of alternating optimization: first, for a fixed transformation $\mathbf{T}$, find the closest correspondences x^*, y^*. Second, fixing the correspondences x^*, y^* find the transformation $\mathbf{T}$ minimizing $D_{\mathrm{H}}(X, \mathbf{T}Y)$.

Such methods are known as *iterative closest point* (ICP). In practice, it is preferable to use a more robust L_2-version of a shape-to-shape distance,

$$D(X,Y) = \int_X \|x - y^*(x)\|^2 dx + \int_Y \|y - x^*(y)\|^2 dy,$$

or a non-symmetric version thereof.

When the shapes X and Y are only partially matching (i.e., there exist unknown in advance parts $X' \subset X, Y' \subset Y$ and a transformation $\mathbf{T}$ such that $D(X', \mathbf{T}Y') \approx 0$), the above method can be adapted by introducing weighting into the shape-to-shape distance,

$$D(X,Y) = \int_X \|x - y^*(x)\|^2 w(x) dx.$$

The weight is set to reject "bad" correspondences, e.g.,

$$w(x) = \begin{cases} 1 & \|x - y^*(x)\| < \epsilon \\ 0 & \text{else} \end{cases}$$

thus effectively excluding the non-overlapping parts of X and Y (rejection can also be made using additional criteria such as angle between normals). Modifying the threshold ϵ implicitly changes the area of the matched parts; however, there is no explicit control of their regularity and area. To overcome this problem, in [5] it was proposed to simultaneously optimize the part dissimilarity, area, and regularity over the parts and the transformation,

$$\min_{\mathbf{T}\in \mathrm{SE}(3), X', Y'} D(X', \mathbf{T}Y') - (A(X') + A(Y')) + (R(X') + R(Y')) \tag{2}$$

where $A(X')$ denotes the area and $R(X')$ the irregularity (e.g. boundary length) of part X'. The purpose of this paper is to extend this idea to multiple part matching, as described in the following.

3 Regularized Multi-part Shape Matching

In the simplest multi-part setting, we have the reference shape X and its unaligned non-overlapping parts $Y_1, \ldots, Y_n$. we assume an initial coarse alignment (e.g. using some global registration); devising a dedicated initialization scheme is deferred to future work. The goal is to match the parts to the references by means of rigid transformations $\mathbf{T}_1, \ldots, \mathbf{T}_n$ such that the matching regions $X_1, \ldots, X_n$ on X are non-overlapping, cover the whole X, regular, and match the area of Y_i (in Sections 3.1 and 3.2, we consider the extension when not all X is covered, and when the parts are cluttered).

The above problem of simultaneous multi-part registration and segmentation can be formulated as

$$\min_{\substack{\{\mathbf{T}_i \in \mathrm{SE}(3)\} \\ \{X_i \subseteq X\}}} \sum_{i=1}^{n} D(X_i, \mathbf{T}_i Y_i) + \lambda \sum_{i=1}^{n} R(X_i) \quad \text{s.t.} \quad \begin{cases} X_i \cap X_j = \emptyset, \ i \neq j \\ X_1 \cup \cdots \cup X_n = X \\ A(X_i) = A(Y_i), \end{cases} \tag{3}$$

where the first aggregate constitutes the data term measuring the proximity of the transformed parts $\mathbf{T}_i Y_i$ to the corresponding segments X_i on the model, while the second aggregate is the regularization term measuring the irregularity of each segment. The first two constraint guarantee that $\{X_i\}$ is a valid partitioning of X, that is, a covering of the latter by disjoint sets. The area constraints ensure that the areas of the segments X_i selected on the model match those of the correspoding parts Y_i.

In order to prevent the segmentation from producing fragmented and irregular segments, we penalize for their boundary length, setting $R(X_i) = L(\partial X_i)$. The discretization of the above problem results in a combinatorial complexity. To circumvent this difficulty, the problem can be relaxed by replacing the crisp parts X_i by fuzzy membership functions u_i on X, and the functional (3) by a generalization of the Mumford-Shah functional [16] to surfaces [4,5]. Here, we adopt this relaxation as well as the Ambrosio-Tortorelli [1] approximation of the Mumford-Shah functional,

$$R(u;\rho) = \frac{\lambda_{\mathrm{s}}}{2}\int_X \rho^2 \|\nabla u\|^2 da + \lambda_{\mathrm{b}}\epsilon \int_X \|\nabla \rho\|^2 da + \frac{\lambda_{\mathrm{b}}}{4\epsilon}\int_X (1-\rho)^2 da, \tag{4}$$

where ρ is the *phase field* indicating the discontinuities of u, and $\epsilon > 0$ is a parameter. The first term of R above imposes piece-wise smoothness of the fuzzy part u. By setting a sufficiently large λ_{s}, the parts become approximately piece-wise constant as desired in the original crisp formulation (3). The second term of R is analogous to the segment boundary length and converges to the latter as $\epsilon \to 0$.

Using this fuzzy formulation, the data term for each u is expressed as

$$D(u, \mathbf{T}Y) = \int_X \|\mathbf{T}\mathbf{y}^*(\mathbf{x}) - \mathbf{x}\|^2 u(\mathbf{x}) da, \tag{5}$$

where Y denotes the corresponding part, one of the Y_i's, and $\mathbf{T}$ its transformation, one of the $\mathbf{T}_i$'s. Combining the data and the regularization terms, we rewrite problem (3) as

$$\min_{\substack{\{\mathbf{T}_i \in \mathrm{SE}(3)\} \\ \{u_i \geq 0, \rho_i \geq 0\}}} \sum_{i=1}^n D(u_i, \mathbf{T}_i Y_i) + \sum_{i=1}^n R(u_i, \rho_i) \quad \text{s.t.} \quad \begin{cases} \sum_{i=1}^n u_i = 1 \\ \int_X u_i da = \int_{Y_i} da. \end{cases} \tag{6}$$

The optimization is performed over n Euclidean transformations $\mathbf{T}_i$, n indicator functions u_i, and corresponding n phase fields ρ_i. The first constraint ensures that the segments u_i constitute a fuzzy partitioning of X and are defined for each point x on X. The rest of the constraints are the fuzzy counterparts of the crisp area constraints in (3).

Missing Parts. In many practical settings, the observed parts Y_i might not cover X entirely e.g. due to occlusions during the acquiring of the objects. In order to handle this scenario, an indicator function u_0 of a "null segment" is added to problem (6). Not corresponding to any of the Y_i's, the null segment is

not subject to area constraints and has no data term; however, it does have a regularity term $R(u_0, \rho_0)$ which is added to the objective. Finally, since the null segment complements the true segments, $u_1, \ldots, u_n$, the point-wise constraint is modified to $\sum_{i=0}^{n} u_i = 1$.

Cluttered Parts. Other practical scenarios may involve the parts Y_i contaminated by clutter, that is, containing foreign objects unrelated to X. We can therefore formulate a partial matching problem by looking for sub-parts $Z_i \subseteq Y_i$ and corresponding segments $X_i \subseteq X$ covering a part of X. In order to address this setting, the previous optimization problem is further extended by adding another set of variables, the indicator functions v_i and the corresponding phase fields σ_i on the Y_i's,

$$\min_{\substack{\{\mathbf{T}_i \in \mathrm{SE}(3)\} \\ \{u_i \geq 0, \rho_i \geq 0\} \\ \{v_i \geq 0, \sigma_i \geq 0\}}} \sum_{i=1}^{n} D(u_i, v_i, \mathbf{T}_i Y_i) + \sum_{i=0}^{n} R(u_i, \rho_i) + \sum_{i=1}^{n} R(v_i, \sigma_i)$$

$$\text{s.t.} \quad \begin{cases} \sum_{i=0}^{n} u_i = 1; \quad v_i \leq 1 \\ \int_X u_i da = \int_{Y_i} v_i da \geq \alpha_i. \end{cases} \tag{7}$$

Note that we still enforce the area constraint, demanding that the area of at least A_i is selected from each Y_i. The latter is a parameter, which may be selected to be either absolute or relative, $\alpha_i = \alpha A(Y_i)$. The data term becomes

$$\begin{aligned} D(u, v, \mathbf{T}Y) = &\frac{1}{2} \int_X \|\mathbf{T}\mathbf{y}^*(\mathbf{x}) - \mathbf{x}\|^2 u(\mathbf{x}) v(\mathbf{y}^*(\mathbf{x})) da(\mathbf{x}) + \\ &\frac{1}{2} \int_Y \|\mathbf{T}\mathbf{y} - \mathbf{x}^*(\mathbf{y})\|^2 u(\mathbf{y}) v(\mathbf{x}^*(\mathbf{y})) da(\mathbf{y}). \end{aligned} \tag{8}$$

While in some applications one can assume reasonable knowledge of α_i, these parameters are often difficult to decide upon *a priori*. As an alternative, we propose reformulate the matching problem with the data term in the constraint, aiming at finding the largest area of the segments (or, equivalently, the smallest area of the null segment) producing a bounded alignment error:

$$\min_{\substack{\{\mathbf{T}_i \in \mathrm{SE}(3)\} \\ \{u_i \geq 0, \rho_i \geq 0\} \\ \{v_i \geq 0, \sigma_i \geq 0\}}} \int_X u_0 da + \sum_{i=0}^{n} R(u_i, \rho_i) + \sum_{i=1}^{n} R(v_i, \sigma_i)$$

$$\text{s.t.} \quad \begin{cases} \sum_{i=0}^{n} u_i = 1; \quad v_i \leq 1 \\ \frac{1}{\beta^2} D(u_i, v_i, \mathbf{T}_i Y_i) \leq \int_X u_i da = \int_{Y_i} v_i da \end{cases} \tag{9}$$

The parameter β^2 controls the maximum allowed mean squared error.

Both problems (7) and (9) are non-convex, yet can be viewed as iteratively reweighted ICP. Indeed, by fixing the u_i's and v_i's, the optimization boils down to solving n weighted rigid matching problems

$$\min_{\{\mathbf{T}_i \in \mathrm{SE}(3)\}} D(u_i, v_i, \mathbf{T}_i Y_i). \tag{10}$$

Next, $\mathbf{T}_i$ are fixed and u_i and v_i and the corresponding ρ_i and σ_i are updated. The process is repeated until convergence. Further details of this alternating minimization algorithm are developed in the sequel.

4 Discretization and Numerical Aspects

We represent the surface X as triangular mesh constructed upon the samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ and denote by $\mathbf{a} = (a_1, \ldots, a_m)^{\mathrm{T}}$ the corresponding area elements at each vertex (the computation of the a_i's is described later). $\mathbf{A} = \mathrm{diag}\{\mathbf{a}\}$ denote the diagonal $m \times m$ matrix created out of $\mathbf{a}$. Each membership function u_i is sampled at each vertex and represented as the vector $\mathbf{u}_i = (u_1^i, \ldots, u_m^i)^{\mathrm{T}}$. Similarly, the phase field is represented as the vector $\boldsymbol{\rho}_i = (\rho_1^i, \ldots, \rho_m^i)^{\mathrm{T}}$. Whenever possible, we will omit the index i to simplify notation. Each of the parts Y_i is represented in the same way as a mesh constructed upon the samples $\{\mathbf{y}_1^i, \ldots, \mathbf{y}_{n_i}^i\}$. The area elements are denoted by $\mathbf{b}_i = (b_1^i, \ldots, b_{n_i}^i)^{\mathrm{T}}$; the membership and the phase field vectors are denoted by $\mathbf{v}_i$ and $\boldsymbol{\sigma}_i$, respectively.

Data Term. For a given part Y (one of the Y_i's), let $\mathbf{y}_i^*$ denote the point corresponding to $\mathbf{x}_i$. The alignment error can be written as $\mathbf{e} = (\|\mathbf{x}_1 - \mathbf{y}_1^*\|^2, \ldots, \|\mathbf{x}_m - \mathbf{y}_m^*\|^2)^{\mathrm{T}}$, and the data term for one part as $D(\mathbf{u}, Y) = \mathbf{u}^{\mathrm{T}}\mathbf{e}$.

Regularization Term. We start by deriving the discretization of a single term $\rho^2 \|\nabla u\|^2 da$ at some point $\mathbf{x}_i$ on the shape. Let us denote by $\mathcal{N}(\mathbf{x}_i)$ the 1-ring of $\mathbf{x}_i$ formed by t vertices $\mathbf{x}_1, \ldots, \mathbf{x}_t$ ordered e.g. in clock-wise order (to simplify notation, we assume without loss of generality consecutive indices). We pick some j-th triangle in $\mathcal{N}(\mathbf{x}_i)$ formed by the central vertex $\mathbf{x}_i$ and the vertices $\mathbf{x}_j$ and $\mathbf{x}_k$ for $k = j \bmod t+1$. Let us denote by $\mathbf{X}_j = (\mathbf{x}_j - \mathbf{x}_i, \mathbf{x}_k - \mathbf{x}_i)$ the 3×2 matrix whose columns are the vectors forming the triangle, and by $\alpha_j = \frac{1}{2}\sqrt{\det(\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)}$ its area. Let also $\mathbf{D}_j$ be the sparse $2 \times m$ matrix with $+1$ at indices $(1, j)$ and $(2, k)$, and -1 at $(1, i)$ and $(2, i)$. $\mathbf{D}_j$ is constructed in such a way to give the differences of values of u on the vertices of the triangle with respect to the values at the central vertex, $\mathbf{D}_j\mathbf{u} = (u_j - u_i, u_k - u_i)^{\mathrm{T}}$. Here, $\mathbf{u}$ denotes one of the membership vectors, $\mathbf{u}_i$. The gradient of the function u is constant on the triangle and can be expressed in these terms by $\mathbf{g}_j = (\mathbf{X}_j^{\mathrm{T}}\mathbf{X}_j)^{-1/2}\mathbf{D}_j\mathbf{u} = \mathbf{E}_j\mathbf{u}$. The area element corresponding to $\mathbf{x}_i$ is given by $a_i = \frac{1}{3}(\alpha_1 + \cdots + \alpha_t)$, and the gradient at that vertex can be expressed by averaging the $\mathbf{g}_j$'s with the weights α_j. This yields

$$\nabla u \, da \approx \frac{1}{3}\sum_{j=1}^{t} \alpha_j \mathbf{g}_j = \frac{1}{3}\sum_{j=1}^{t} \alpha_j \mathbf{E}_j\mathbf{u} = \frac{1}{3}((\alpha_1, \ldots, \alpha_t) \otimes \mathbf{I})\mathbf{E}\mathbf{u}, \tag{11}$$

where $\mathbf{1}$ is a $2t \times 1$ vector of ones, $\mathbf{E}$ is the $2t \times m$ matrix stacking $\mathbf{E}_j$'s, $\mathbf{I}$ is the 2×2 identity matrix, and $\otimes$ denotes the Kroenecker product $(\alpha_1, \dots, \alpha_t) \otimes \mathbf{I} = (\alpha_1 \mathbf{I}, \dots, \alpha_t \mathbf{I})$. Denoting by $\mathbf{G}_i$ the $2 \times m$ matrix $\frac{1}{3}((\alpha_1, \dots, \alpha_t) \otimes \mathbf{I})\mathbf{E}$ corresponding to the vertex $\mathbf{x}_i$, we can write $\nabla u \, da \approx \mathbf{G}_i \mathbf{u}$.

Let us now consider all the points of the shape. We have

$$\int_X \rho^2 \|\nabla u\|^2 da \approx \sum_{i=1}^{m} \frac{\rho_i^2}{a_i} \mathbf{u}^{\mathrm{T}} \mathbf{G}_i^{\mathrm{T}} \mathbf{G}_i \mathbf{u}. \tag{12}$$

Introducing a $2m \times m$ matrix

$$\mathbf{G} = \left(\mathrm{diag} \left\{ \frac{1}{\sqrt{a_1}}, \dots, \frac{1}{\sqrt{a_m}} \right\} \otimes \mathbf{I} \right) \begin{pmatrix} \mathbf{G}_1 \\ \vdots \\ \mathbf{G}_m \end{pmatrix} \tag{13}$$

allows to rewrite the former integral as

$$\int_X \rho^2 \|\nabla u\|^2 da \approx \|(\mathrm{diag}\{\boldsymbol{\rho}\} \otimes \mathbf{I})\mathbf{G}\mathbf{u}\|^2 = \mathbf{u}^{\mathrm{T}} \mathbf{G}^{\mathrm{T}} (\mathrm{diag}\{\boldsymbol{\rho}^2\} \otimes \mathbf{I}) \mathbf{G}\mathbf{u}. \tag{14}$$

Similarly,

$$\int_X \|\nabla \rho\|^2 da \approx \|\mathbf{G}\boldsymbol{\rho}\|^2 = \boldsymbol{\rho}^{\mathrm{T}} \mathbf{G}^{\mathrm{T}} \mathbf{G} \boldsymbol{\rho} \tag{15}$$

$$\int_X (1 - \rho)^2 da \approx \boldsymbol{\rho}^{\mathrm{T}} \mathbf{A} \boldsymbol{\rho} - 2\mathbf{a}^{\mathrm{T}} \boldsymbol{\rho} + \mathbf{1}^{\mathrm{T}} \mathbf{a}. \tag{16}$$

The discretized regularization term

$$\begin{aligned} R(\mathbf{u}, \boldsymbol{\rho}) = \frac{\lambda_{\mathrm{s}}}{2} \mathbf{u}^{\mathrm{T}} \mathbf{G}^{\mathrm{T}} (\mathrm{diag}\{\boldsymbol{\rho}^2\} \otimes \mathbf{I}) \mathbf{G}\mathbf{u} + \\ \lambda_{\mathrm{b}} \boldsymbol{\rho}^{\mathrm{T}} \left(\epsilon \mathbf{G}^{\mathrm{T}} \mathbf{G} + \frac{1}{4\epsilon} \mathbf{A} \right) \boldsymbol{\rho} - \frac{\lambda_{\mathrm{b}}}{4\epsilon} \left(2\mathbf{a}^{\mathrm{T}} \boldsymbol{\rho} + \mathbf{1}^{\mathrm{T}} \mathbf{a} \right) \end{aligned} \tag{17}$$

is, therefore, quadratic in $\mathbf{u}$ and $\boldsymbol{\rho}$ independently (but not in both simultaneously!). Regularization terms for $\mathbf{v}_i$ and $\boldsymbol{\sigma}_i$ are obtained in the same manner.

Alternating Minimization. The problem (6) is solved by means of alternating minimization, in the following steps:

1. Fix $\{\mathbf{u}_i, \boldsymbol{\rho}_i\}$ and compute the transformation $\{\mathbf{T}_i\}$ minimizing the data term $\sum_{i=1}^{n} D(\mathbf{u}_i, \mathbf{T}_i Y_i)$. This step is akin to a step of multiple individual weighted ICP problems.
2. Update $Y_i = \mathbf{T}_i Y_i$, compute the correspondence and the alignment errors $\mathbf{e}_i$.
3. Fix $\{\mathbf{T}_i, \mathbf{e}_i, \boldsymbol{\rho}_i\}$ and compute the weights $\{\mathbf{u}_i\}$ solving the *quadratic programming* (QP) problem

$$\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{n} \mathbf{u}_i^{\mathrm{T}} \mathbf{e}_i + \frac{\lambda_s}{2} \mathbf{u}_i^{\mathrm{T}} \mathbf{B}_i \mathbf{u}_i \;\; \text{s.t.} \;\; \sum_{i=1}^{n} \mathbf{u}_i = 1; \;\; \mathbf{a}^{\mathbf{T}} \mathbf{u}_i = A(Y_i); \;\; \mathbf{u}_i \geq 0$$

   where $\mathbf{B} = \mathbf{G}^{\mathrm{T}} (\mathrm{diag}\{\boldsymbol{\rho}^2\} \otimes \mathbf{I}) \mathbf{G}$ as defined in (17).
4. Fix $\{\mathbf{T}_i, \mathbf{e}_i, \mathbf{u}_i\}$ and compute the phase fields $\{\boldsymbol{\rho}_i\}$ by solving the unconstrained optimization problem

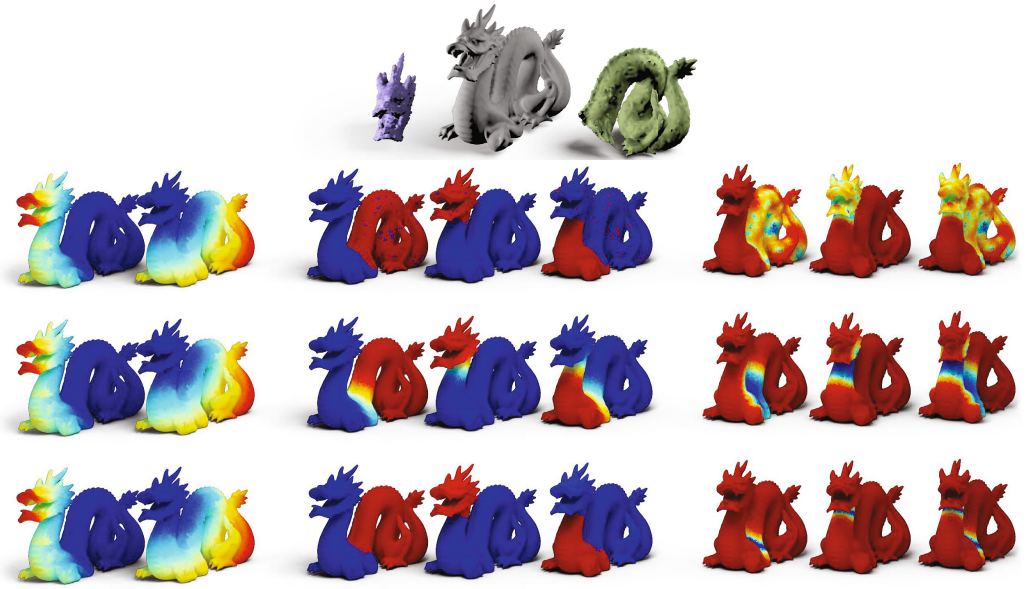

Fig. 2. Multi-part marching with missing parts. First row: two parts and the reference shape; Second row: initialization with ICP; Third row: first iteration of alternate minimization; Fourth row: final result. Columns 1-2: data term; columns 3-5: u_i, columns 6-8: ρ_i.

$$\min_{\{\boldsymbol{\rho}_i\}} \sum_{i=1}^{n} \boldsymbol{\rho}_i^{\mathrm{T}} \mathbf{C}_i \boldsymbol{\rho}_i - \frac{\lambda_{\mathrm{b}}}{2\epsilon} \mathbf{a}^{\mathrm{T}} \boldsymbol{\rho}_i$$

where $\mathbf{C} = \left(\frac{\lambda_{\mathrm{s}}}{2} \mathbf{S}(\mathbf{u}) + \lambda_{\mathrm{b}} \epsilon \mathbf{G}^{\mathrm{T}} \mathbf{G} + \frac{\lambda_{\mathrm{b}}}{4\epsilon} \mathbf{A} \right)$, $\mathbf{S}(\mathbf{u}) = \mathrm{diag}\{s_1, \ldots, s_m\}$ and $s_i = \frac{1}{a_i} \mathbf{u}^{\mathrm{T}} \mathbf{G}_i^{\mathrm{T}} \mathbf{G}_i \mathbf{u} \approx \|\nabla u\|^2 da$ at vertex $\mathbf{x}_i$. The solution for each $\boldsymbol{\rho}_i$ is given by

$$\boldsymbol{\rho} = \left(2 \frac{\lambda_{\mathrm{s}} \epsilon}{\lambda_{\mathrm{b}}} \mathbf{S}(\mathbf{u}) + 4\epsilon^2 \mathbf{G}^{\mathrm{T}} \mathbf{G} + \mathbf{A} \right)^{-1} \mathbf{a}.$$

Problems (7)–(9)) are solved in a similar way: we fix all the parameters above, compute the correspondence and follow steps 3 and 4 for $\{\mathbf{v}_i\}$ and $\{\boldsymbol{\sigma}_i\}$, respectively.

5 Experimental Results

In this section, we show the performance of our algorithm under different settings on three shapes from the Stanford repository (bunny, armadillo, and dragon). The algorithm was implemented in MATLAB. Execution time depended on the number of vertices and parts; typical execution time for a reference shape with 10^3 points and 5 parts was 3.5 sec. Figure 2 illustrates the different stages of our algorithm on the problem of matching two parts (head and tail) of the dragon shape with a missing part (chest).

Effect of Regularization. is shown in Figure 3. In this experiment, we use the armadillo model fragmented into 4 parts, contaminated by Gaussian noise. The resulting segmentation of the reference model varies with the modification of

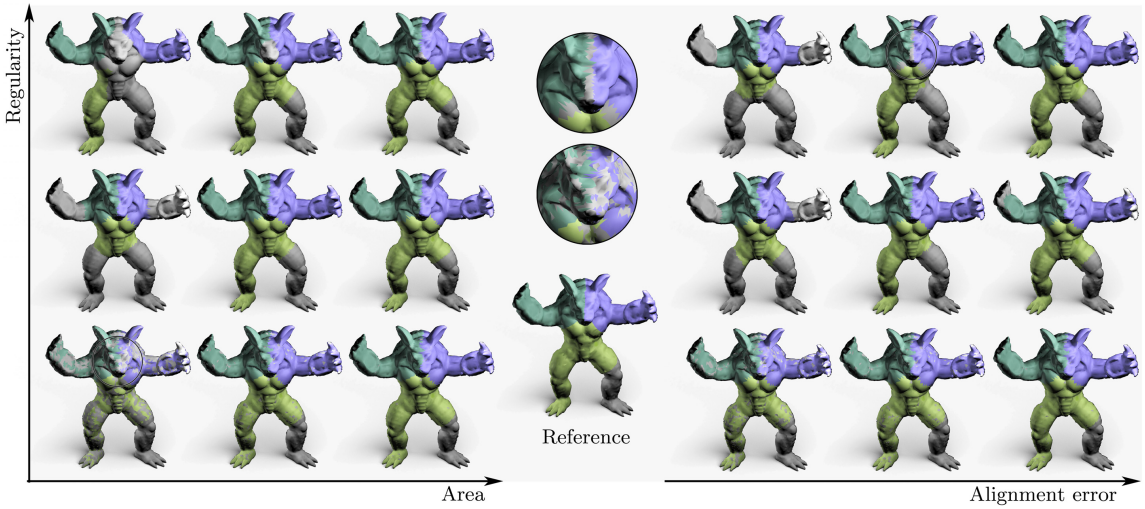

Fig. 3. The effect of different terms and constraints in the problem. Note the effect of over-regularization which causes shortening of the segment boundaries (top close-up), and under-regularization causing fragmented segments (bottom close-up).

Fig. 4. Noisy parts of the dragon shape (left) and the matching result (right) using the area constraint setting. The reference shape is shown in gray.

the area (Figure 3, left) and alignment error (Figure 3, right) constraints. Small area constraint results in large portions of the shape marked as "missing part". A similar result is obtained when allowing small alignment error. Increasing the regularity penalty encourages segmentation into parts with smoother boundaries.

Handling Noise. is shown in Figure 4, where we fragmented the dragon shape into four parts contaminated by gaussian noise and matched them to the clean reference shape using our approach (the area constraint was computed on the clean parts).

Handling Clutter. is shown in Figure 1. In this experiment, we printed the bunny shape fragmented into ten parts using a 3D printer. Leaving one part (the right ear) out, we scanned the remaining parts using a 3D scanner. The scan imperfections are clearly seen as noise, clutter, and holes. Solving our problem with the error in the constraint for indicators on the model and on each of the parts, we get the segmentation shown in Figure 1 (center). Having removed the clutter in this way, the segmented fragments fit together correction (Figure 1, right).

6 Conclusions

We presented an efficient alternate optimization scheme for solving simultaneous registration and segmentation of multi-part shapes. We are not aware of any other method which handle both problems simultaneously. Our approach can handle noise, clutter, and missing part, as shown on real 3D data examples. In future works, we will extend our method to the matching of solid (volumetric) 3D shapes, such as fragments encountered in archaeological applications.

References

1. Ambrosio, L., Tortorelli, V.M.: Approximation of functional depending on jumps by elliptic functional via t-convergence. Communications on Pure and Applied Mathematics 43(8), 999–1036 (1990)
2. Besl, P.J., McKay, N.D.: A method for registration of 3D shapes. Trans. PAMI 14(2), 239–256 (1992)
3. Bronstein, A., Bronstein, M., Bruckstein, A., Kimmel, R.: Partial similarity of objects, or how to compare a centaur to a horse. IJCV 84(2), 163–183 (2009)
4. Bronstein, A.M., Bronstein, M.M.: Not only size matters: regularized partial matching of nonrigid shapes. In: Proc. NORDIA (2008)
5. Bronstein, A.M., Bronstein, M.M.: Regularized Partial Matching of Rigid Shapes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 143–154. Springer, Heidelberg (2008)
6. Chen, Y., Medioni, G.: Object modeling by registration of multiple range images. In: Proc. Conf. Robotics and Automation (1991)
7. Domokos, C., Kato, Z.: Affine Puzzle: Realigning Deformed Object Fragments without Correspondences. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 777–790. Springer, Heidelberg (2010)
8. Gelfand, N., Mitra, N.J., Guibas, L.J., Pottmann, H.: Robust global registration. In: Proc. SGP (2005)
9. Hori, K., Imai, M., Ogasawara, T.: Joint detection for potsherds of broken earthenware. In: Proc. CVPR (1999)
10. Huang, Q.X., Flöry, S., Gelfand, N., Hofer, M., Pottmann, H.: Reassembling fractured objects by geometric matching. In: TOG, vol. 25, pp. 569–578 (2006)
11. Huber, D.F.: Automatic three-dimensional modeling from reality. PhD thesis, Carnegy Mellon University (2002)
12. Inbar, Y., Wolfson, H.J., Nussinov, R.: Multiple docking for protein structure prediction. International Journal of Robotics Research 24(2-3), 131–150 (2005)
13. Kong, W., Kimia, B.B.: On solving 2d and 3d puzzles using curve matching. In: Proc. CVPR (2001)
14. Krishnan, S., Lee, P.Y., Moore, J.B., Venkatasubramanian, S.: Global registration of multiple 3d point sets via optimization-on-a-manifold. In: Proc. SGP (2005)
15. Mitra, N.J., Gelfand, N., Pottmann, H., Guibas, L.J.: Registration of point cloud data from a geometric optimization perspective. In: Proc. Eurographics Symposium on Geometry Processing, pp. 23–32 (2004)
16. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics 42(5), 577–685 (1989)

17. Neugebauer, P.J.: Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images. International Journal of Shape Modeling 3, 71–90 (1997)
18. Papaioannou, G., Karabassi, E.A.: On the automatic assemblage of arbitrary broken solid artefacts. Image and Vision Computing 21(5), 401–412 (2003)
19. Papaioannou, G., Karabassi, E.A., Theoharis, T.: Virtual archaeologist: Assembling the past. Computer Graphics and Applications 21(2), 53–59 (2001)
20. Pokrass, J., Bronstein, A.M., Bronstein, M.M.: A correspondence-less approach to matching of deformable shapes. In: Proc. SSVM (2012)
21. Pulli, K.: Multiview registration for large data sets. In: Prof. Conf. 3D Digital Imaging and Modeling (1999)