

Field-Guided Registration for Feature-Conforming Shape Composition

Hui Huang* Minglun Gong† Daniel Cohen-Or‡ Yaobin Ouyang* Fuwen Tan* Hao Zhang§
*Shenzhen VisuCA Key Lab / SIAT †Memorial Univ. of Newfoundland ‡Tel-Aviv Univ. §Simon Fraser Univ.

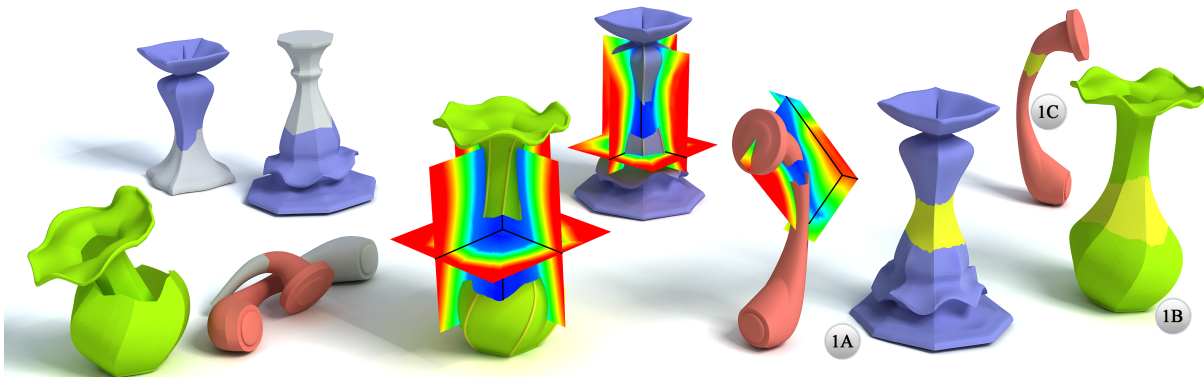


Figure 1: Composing parts, possibly with sharp features and non-overlapping boundaries, presents challenges to both part alignment and blending. Our field-guided approach (see middle for a visualization of the fields) leads to alignment of parts away from each other and feature-conforming surface blending. The bridging surfaces generated (colored yellow on the right) are piecewise smooth.

Abstract

We present an automatic shape composition method to fuse two shape parts which may not overlap and possibly contain sharp features, a scenario often encountered when modeling man-made objects. At the core of our method is a novel *field-guided* approach to automatically align two input parts in a *feature-conforming* manner. The key to our field-guided shape registration is a *natural continuation* of one part into the ambient field as a means to introduce an overlap with the distant part, which then allows a surface-to-field registration. The ambient vector field we compute is feature-conforming; it characterizes a piecewise smooth field which respects and naturally extrapolates the surface features. Once the two parts are aligned, gap filling is carried out by spline interpolation between matching feature curves followed by piecewise smooth least-squares surface reconstruction. We apply our algorithm to obtain feature-conforming shape composition on a variety of models and demonstrate generality of the method with results on parts with or without overlap and with or without salient features.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—[Curve, surface, solid, and object representations]

Keywords: field-guided registration, feature-conforming, shape composition, interpolation, extrapolation, piecewise smooth

Links:  DL  PDF  WEB

ACM Reference Format
Huang, H., Gong, M., Cohen-Or, D., Ouyang, Y., Tan, F., Zhang, H. 2012. Field-Guided Registration for Feature-Conforming Shape Composition. *ACM Trans. Graph.* 31 6, Article 179 (November 2012), 11 pages. DOI = 10.1145/2366145.2366198 <http://doi.acm.org/10.1145/2366145.2366198>.

Copyright Notice
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701, fax +1 (212) 869-0481, or permissions@acm.org.
© 2012 ACM 0730-0301/2012/11-ART 179 \$15.00 DOI 10.1145/2366145.2366198
<http://doi.acm.org/10.1145/2366145.2366198>

1 Introduction

Model creation in 3D is one of the most central problems in computer graphics. With the rapid growth of online shape repositories, advanced approaches to shape creation have opted to exploit the large number of available models. Perhaps the most widely applied modeling paradigm is to create or modify shapes by means of composing parts belonging to existing shapes [Funkhouser et al. 2004]. Often, the composing parts need to be fused together in a conforming, seamless, and coherent manner so that the resulting shapes can be readily used for subsequent processing. Despite much work on shape composition with a focus on high-level tasks [Funkhouser et al. 2004; Kreavoy et al. 2007; Chaudhuri et al. 2011; Kalogerakis et al. 2012; Xu et al. 2012], the low-level part composition problem has received considerably less attention.

The process of composing two parts together consists of two sub-problems: (i) *alignment*: the two parts should be properly aligned spatially to allow a coherent composition; and (ii) *blending*: the surface(s) connecting the two parts should conform gracefully to the part geometry. These two problems are inter-related as clearly a good alignment simplifies the blending, and a powerful blending compensates for an imperfect alignment.

Most works on shape composition require a user to bring the parts into the right positions and pose [Yu et al. 2004; Sorkine et al. 2004; Kreavoy et al. 2007; Schmidt and Singh 2010]; this can be a delicate task when the parts contain sharp features to be aligned. Methods developed for automatic part alignment have appeared mostly in the shape registration literature [van Kaick et al. 2011], not shape composition. The major difference between the two is that registration almost always assumes and relies on the two parts to sufficiently overlap [Rusinkiewicz and Levoy 2001; Gelfand et al. 2005; Aiger et al. 2008], which is typically not the case for shape composition since the parts originate from different models.

Existing approaches to both part alignment and blending have all been designed to handle smooth geometry [Yu et al. 2004; Sorkine et al. 2004; Kreavoy et al. 2007; Sharf et al. 2006; Lin et al. 2008; Schmidt and Singh 2010]. In the presence of sharp features, which are ubiquitous in man-made models, the composition problem be-

comes more challenging. First, the shape features on both parts necessitate feature alignment and a smooth conformation between the features. Second, in the absence of an exact feature matching, feature alignment is not straightforward. Last but not least, gap filling and part blending must respect the sharp features, which necessitates a piecewise smooth formulation.

In this paper, we present an *automatic* method for composing two shape parts which may not overlap, with each possibly containing sharp features. The part alignment problem corresponds to a new kind of registration, namely, registering parts which may be away from each other. Our key idea is to construct a natural part continuation to introduce an overlap as necessary and doing so in the *ambient field* instead of explicitly extending the part surfaces. Such a field-based approach provides a more flexible and effective means to deal with dissimilarity between the parts, which often come from different origins. We call our algorithm *field-guided registration*. Specifically, the parts are registered by first computing an ambient vector field continuation from one part. Then the surface of the other part is brought to alignment with respect to the extension field by sampling and searching for the proper part pose.

The ambient vector field computed is a *feature-conforming field* (FCF), which enables feature-conforming part alignment and blending. The FCF characterizes a piecewise smooth volume that respects and naturally extrapolates the sharp features on the source surface. The surface-to-field registration is also feature-conforming as it obeys the surface orientations. Once the two parts are well aligned, gap-filling is carried out by spline interpolation between matching feature curves followed by piecewise smooth surface reconstruction [Mallet 1989; Sorkine and Cohen-Or 2004].

Contributions. Our main contributions are:

- A field-guided registration algorithm that is applicable to non-overlapping shape parts; this is enabled by an ambient field continuation and a surface-to-field registration.
- A feature-conforming vector field construction, which respects and extrapolates a piecewise smooth input surface.
- A feature-conforming shape composition, allowing a seamless assembly of parts even with sharp features.

Figure 1 shows a first glimpse of our shape composition results, where the shape parts are automatically aligned without overlap and the gap is filled in a piecewise smooth manner. The field-guided registration framework is quite general and capable of composing shapes with or without salient features and with or without overlaps, as we demonstrate on a variety of 3D models.

2 Related Works

Shape composition has been investigated in various forms in computer graphics. Several works extend the notion of drag-and-drop or image cloning to surfaces [Biermann et al. 2002; Fu et al. 2004; Sorkine et al. 2004; Takayama et al. 2011], where geometric details with a disc topology are pasted onto a surface. Other works focus on smooth blending between the boundaries of two composed parts [Singh and Parent 2001; Museth et al. 2002; Yu et al. 2004; Huang et al. 2007; Lin et al. 2008; Schmidt and Singh 2010]. In these methods, part positioning and pose are defined by the user, while in our work, we focus on automatic part alignment.

Composing shape parts for model creation is mostly carried out in an interactive setting. However, to precisely position the parts in 3D space for an optimal part fusion can be a tedious task for the user, especially when the parts possess features to be matched up. Sharf

et al. [2007] rely on a soft ICP registration to automatically snap the manipulated part to its proper position as soon as the part is brought to overlap with the composition target. Our registration also introduces a snap, but the composed parts are not required to overlap and the manipulated part is snapped to an ambient field. Moreover, unlike Sharf et al. [2007], our method is feature-conforming.

Methods which follow the “modeling by example” paradigm of Funkhouser et al. [2004] create 3D models via part composition. The focus of these methods have mostly been on high-level tasks such as ensuring compatibility between replaceable parts [Kreavoy et al. 2007] or identifying the most relevant parts for composition [Chaudhuri et al. 2011; Kalogerakis et al. 2012; Xu et al. 2012]. These methods implicitly suggest that seamless and coherent fusion between composed parts is fundamentally important for creating readily usable models, but do not explicitly address the challenging issues faced by the part fusion problem, e.g., automatic part alignment or feature conformation.

Hassner et al. [2005] present a method to automatically determine the best place to connect two shape parts, under user-specified constraints. However, the optimization of their connection assumes that the two parts are pre-aligned and no transformation is applied to either part. Most alignment and registration methods assume that the parts to be aligned or registered have sufficient overlap [van Kaick et al. 2011]. When that is the case, partial matching techniques [Gelfand et al. 2005; Li and Guskov 2005; Gal and Cohen-Or 2006; Aiger et al. 2008; Itskovich and Tal 2011] can be applied to solve the alignment problem. The surface motion can be expressed in terms of a reduced deformable model [Chang and Zwicker 2009] so that multiple range scans can be aligned and simultaneously reconstructed for a full articulated 3D model [Chang and Zwicker 2011]. Again, our approach does not require any part overlap. Moreover, it is based on a global surface-to-field registration whose result is insensitive to the initial part positions.

Our part alignment scheme is based on a novel surface continuation, which generates an ambient vector field with respect to surface sharp features. Other means to extrapolate 3D surface samples into a volumetric implicit function include radial basis functions (RBF) [Carr et al. 2001], Poisson interpolation [Kazhdan et al. 2006], or the use of finite element (FEM) formulations [Sharf et al. 2007] and dual domain [Lévy 2003]. There are also many works that are based on 3D distance transforms [Jones et al. 2006]. Our FCF-based extension mechanism is most similar to Tensor Voting (TV) [Tang and Medioni 2002; Medioni and Kang 2004], which has also been applied to surface modeling [Beltrowska et al. 2008]. All the aforementioned methods deal with smooth surfaces and generate smooth ambient fields. In contrast, the ambient field we compute is feature-conforming and piecewise smooth.

Although in this paper we only consider the case of registering two objects and piecewise smooth surface extrapolation, it is intriguing to enhance our technique for a cultural heritage application, specifically, to reassemble archaeological artifacts from an incomplete set of pieces [Toler Franklin et al. 2010; Funkhouser et al. 2011]; note that these pieces cannot overlap. The system proposed by Huang et al. [2006] reassembles broken pieces by analyzing their geometry. However, it is based on a global registration of pairwise matching of adjacent pieces, implicitly requiring that the assembly is connected.

3 Overview

Given two parts P and Q represented by triangular meshes, the goal of our algorithm is to align and blend them into a complete piecewise smooth mesh model. We do not assume that the aligned pieces overlap, which allows non-conforming parts to be placed apart so that a smooth transitions between them may be constructed. In

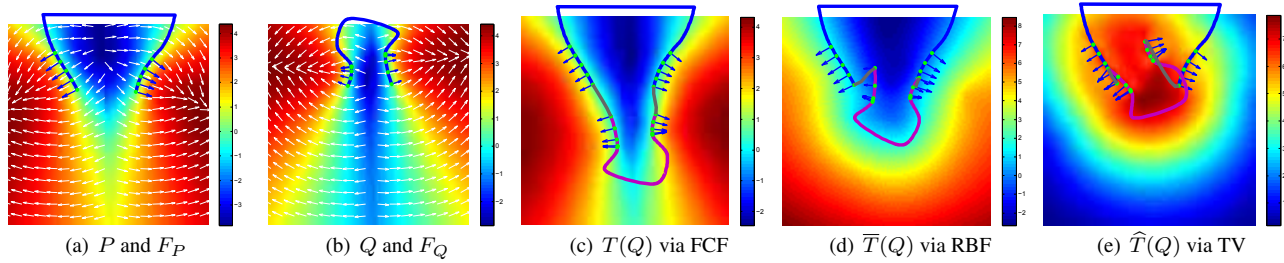


Figure 2: Field-guided registration with three different fields compared: (a) the ambient vector field F_P for part P , where sampled vertices on P are shown as green dots with blue normals; (b) the vector field F_Q of part Q ; (c) our registration result shown on top of the combined field $F_{PQ} = F_P + T(F_Q)$, where T is the aligning rigid transformation and the grey lines are the fitting spline curves between P and $T(Q)$; (d) registering $\bar{T}(Q)$ via RBF fields of both P and Q ; (e) registering $\hat{T}(Q)$ via TV fields of both P and Q .

other words, we search for a rigid transformation that aligns the two parts so that the surface in between behaves as a natural continuation and blend between them. Meanwhile, the sharp features of the two parts are respected by the composition.

Our approach to natural continuation is to extend the influence of each part's surface into the ambient space. With the parts P and Q extrapolated into ambient fields F_P and F_Q , we transform the difficult non-overlapping registration problem into a problem of registering Q to F_P and P to F_Q using a consistent transformation T . Once the two parts are properly aligned, they are connected together via piecewise smooth bridging surface patches.

The key to our field-guided approach is to define an appropriate ambient vector field for each part, which may contain multiple surface patches with sharp edges between them. Unlike existing field generation approaches, such as RBF and TV, which blend together the influence of surface patches facing different directions and thus blur the sharp edges, our field respects sharp features. In fact, both the direction and the magnitude of our vector field are computed using positional and normal information of vertices on surface patches. As shown in Figures 2(a) and 2(b), this allows our field to naturally extend both the surface position and orientation into the ambient space in a piecewise smooth manner. Blending between patches with different directions is avoided and thus sharp features among them are respected. This leads to better registration results than using RBF and TV fields; see Figures 2(c-e).

4 Field-Guided Registration

Our registration technique consists of two major steps. First, we define a vector field F_P for each part P based on a set of vertices Ω_P in the vicinity of P 's open boundaries. Such a field contains information about the possible piecewise smooth extension of P and measures the likelihood and conformance that a surface lay on it. Then, to register two parts P and Q in a feature-conforming manner, we search for a *global optimal* rigid transformation T such that $T(Q)$ matches with F_P and $T^{-1}(P)$ matches with F_Q .

Field generation. This step computes a vector field F_P for a given part P that extends the surface outwards starting from P 's open boundary. Unlike existing fields, such as TV and RBF, which assume that the underlining surface is smooth, our feature-conforming field (FCF) respects sharp features of the surface.

Our goal is to use F_P to compute the likelihood of a given vertex v lying on the extension of P . As shown in Figure 3, the likelihood of v being on the extension of a surface depends not only on the location of v but also on its normal. After all, reorienting a ver-

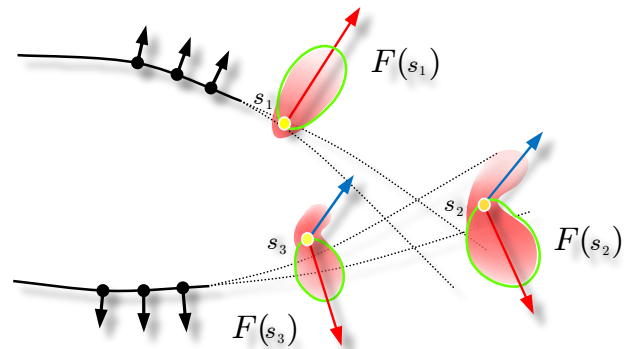


Figure 3: A 2D illustration showing the likelihood of different points lying on surface extensions. Dashed lines show possible extensions of the surface. The desired likelihood values as a function of point's orientation are plotted using polar coordinates at three locations. The optimal direction which yields the maximum likelihood is shown in red. The secondary optimal direction, which may also exist, is shown in blue. Our vector field $F(s)$ is used for approximating the likelihood variation in the neighborhood of the maximum likelihood direction only, i.e., the green curves.

tex can have a large impact on the shape of an interpolating surface. However for 3D models, fully representing the likelihood variations under different locations and normal orientations requires a 5D representation, which leads to expensive computation and storage. To simplify the problem, we use $F_P(s)$ to store the direction that yields the maximum likelihood at location s and a scalar used for computing the corresponding likelihood value. This information allows us to approximate the likelihood variation in the neighborhood of the maximum likelihood direction; see Figure 3.

Since vertices on the surface extension are usually expected to have normals similar to those on the original surface, we can assume that, at a given location s , the maximum likelihood occurs along the direction $\mathbf{d}_P(s)$ that coincides with the normals of nearby vertices. Therefore, to compute $\mathbf{d}_P(s)$, we optimize the following function:

$$\mathbf{d}_P(s) = \max_{\mathbf{d}} \frac{\sum_{u \in \Omega_P} \phi_1(s, u) \phi_2(\mathbf{d}, u)}{\sum_{u \in \Omega_P} \phi_1(s, u)}, \quad (1)$$

$$\phi_1(s, u) = \exp\left(-\frac{(1 - \mathbf{n}_u^\top \frac{(s-u)}{|s-u|})^2}{(1 - \cos \sigma_1)^2}\right),$$

$$\phi_2(\mathbf{d}, u) = \exp\left(-\frac{(1 - \mathbf{n}_u^\top \mathbf{d})^2}{(1 - \cos \sigma_2)^2}\right),$$

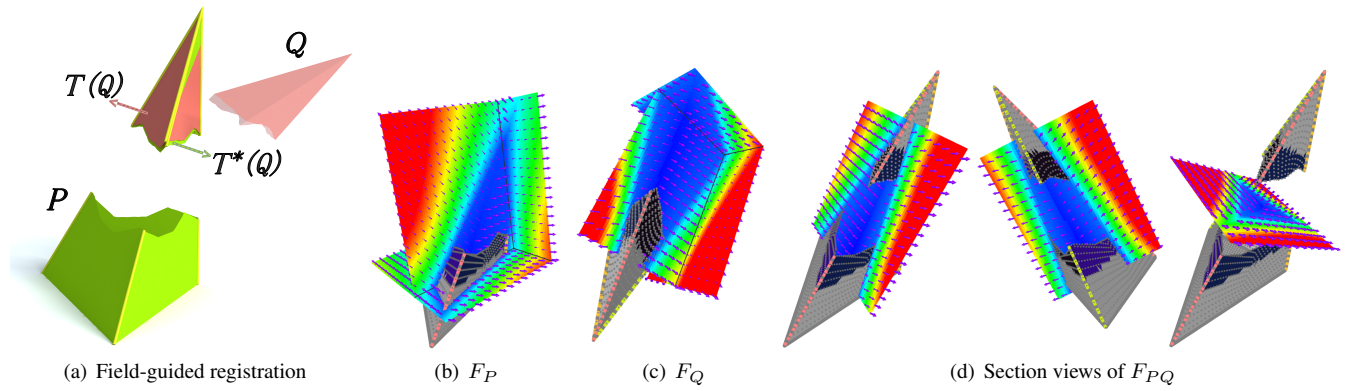


Figure 4: Registration for a synthetic model, whose ground truth transformation T^* is known. A pyramid was cut in the middle, with the top part Q displaced as shown by the light red part in (a). The rigid transform T obtained using our field-guided registration, shown as $T(Q)$ in solid red in (a), has a likelihood score of $E(T) = 0.87$, while the ground truth (shown in green) has the highest score of $E(T^*) = 0.91$. The FCF for the parts and the combine field after registration are shown in (b), (c), and (d), respectively.

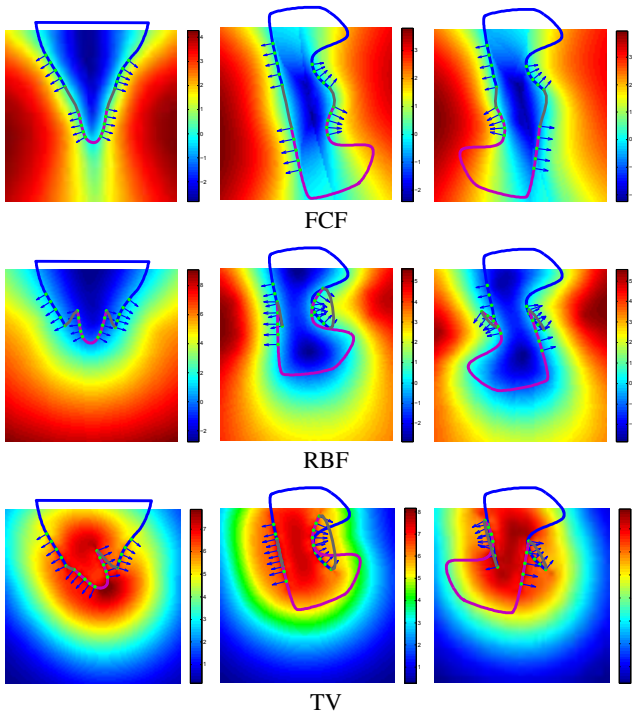


Figure 5: More field-guided registration results in 2D: the top row demonstrates the curves composed within our FCF; the middle and bottom rows present the curves stitched via RBF and TV, respectively. Since the openings of the matching curves are not conforming to each other, RBF and TV fail to properly align the parts. FCF on the other hand places parts apart from each other, leading to smooth and natural connections.

where parameters σ_1 and σ_2 control the supports of two Gaussian functions, with default values 65° and 45° , respectively. Among the two terms, the direction evaluation term $\phi_2(\mathbf{d}, u)$ is the dominating one, which favors the direction \mathbf{d} that matches the normal direction of a given vertex u . The normalized weight term $\phi_1(s, u)$ gives vertices with normal pointing toward location s higher weights. Together, they naturally extend the normal directions of vertices in Ω_P into the ambient space in a piecewise smooth

manner; see Figures 2 and 4.

Next, we compute a signed distance between location s and surface extensions of part P along the direction $\mathbf{d}_P(s)$. This is achieved using the following function

$$l_P(s) = \frac{\sum_{u \in \Omega_P} \phi_2(\mathbf{d}_P(s), u) \phi_3(s, u)}{\sum_{u \in \Omega_P} \phi_2(\mathbf{d}_P(s), u)}, \quad (2)$$

$$\phi_3(s, u) = \frac{\mathbf{n}_u^\top (s - u)}{|s - u| + c}.$$

Here c is a positive constant, which is set to 1 by default. The function $\phi_2(\cdot, \cdot)$ follows the definition in (1), serving as a normal weight term by assigning higher weights to vertices with normals pointing along the maximum likelihood direction $\mathbf{d}_P(s)$. $\phi_3(s, u)$ is an adjusted distance term, whose numerator computes the signed distance from position s to the tangential plane of vertex u and denominator adjusts the signed distance value using the distance between s and u , introducing bias toward positions that are further away from u as they allow the bridging surfaces to have smaller curvatures. Hence, the closer $\phi_3(s, u)$ is to zero, the more likely position s is on the surface extension defined by u .

The direction $\mathbf{d}_P(s)$ and the signed distance $l_P(s)$ together form the FCF $F_P(s)$. As shown in Figures 2 and 5 for a 2D case and Figure 1 and 4 for a 3D case, FCF naturally extends the shapes of the input parts into the ambient space. Compared to RBF and TV, FCF better preserves sharp features.

With FCF defined, the likelihood of vertex v with normal \mathbf{n}_v lying on the surface extension of part P is approximated by:

$$K_P(v) = (\max(\mathbf{n}_v^\top \mathbf{d}_P(v), 0))^w e^{-|l_P(v)|}, \quad (3)$$

where w is a constant parameter (2 by default), whose effect is similar to the roughness parameter used for modeling the specular reflection in Phong illumination model. Evidently, the first term in the above expression is maximized when the vertex normal \mathbf{n}_v coincides with maximum likelihood direction $\mathbf{d}(v)$ and the second term is maximized when vertex v is close to the contour surfaces of ϕ_3 computed using existing vertices. Together, the two terms output a high likelihood value when the vertex v aligns well with the surface extension of part P .

Field-guided rigid alignment. With fields F_P and F_Q defined for parts P and Q , respectively, we perform a field-guided surface-to-field alignment. That is, we solve for a rigid transformation T between both P and Q given by

$$T = \arg \max_T E(T), \quad (4)$$

$$E(T) = \frac{1}{N_Q} \sum_{v \in \Omega_Q} K_P(T \cdot v) + \frac{1}{N_P} \sum_{u \in \Omega_P} K_Q(T^{-1} \cdot u),$$

where $N_P = |\Omega_P|$ and $N_Q = |\Omega_Q|$ are normalization factors. Defining the objective function in such a manner maximizes the total likelihood of the two parts in each other's fields, resulting in a more robust solution. It also ensures that the field-guided alignment is order-independent, i.e., aligning P with Q gives the same result as aligning Q with P .

To compute the rigid transformation T by (4), we utilize the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [Nocedal and Wright 2006]. Unlike the standard Newton optimization, the BFGS scheme approximates the Hessian matrix using rank-two updates specified by gradient evaluations. Our implementation is based on a Matlab function written by Mark Schmidt, called *minFunc*¹. This function uses an interface similar to the *fminunc* function in the Matlab Optimization Toolbox, but converges faster in general and is able to optimize problems with a larger number of variables. Further it allows for numerical gradients and hence only an initial guess is required.

In our experiments, the termination tolerance on the function value is 0.01 and the maximum number of iterations allowed is 500. The gradients are computed numerically using central-differencing. A bracketing line search for a point satisfying the strong Wolfe conditions is used to compute the step size with an initial step length of 1, and cubic interpolation is employed as a safeguard in the line search. All other parameters used to generate all the results in the paper taking the default setting from *minFunc*.

Like Newton techniques, BFGS requires a good initial guess to guarantee convergence to a local maximum, especially in our application where a large number of local maxima may exist. Thus, we employ the random search (RS) strategy [Rastrigin 1963], i.e., we randomly sample the 5D rigid transformation space and pick the sample T that yields the maximum $E(T)$ value as the initial solution. In practice, we found that when the initial solution is chosen from 10K random samples, the above RS+BFGS search consistently outputs the expected rigid transformation.

The field-guided registration algorithm is summarized below:

Input: P, Q, Ω_P and Ω_Q
for Part P do
 Step 1: build a large grid G_P in space that contains Ω_P .
 Step 2: for each cell s of G_P , compute $d_P(s)$ by (1).
 Step 3: for each cell s of G_P , compute $l_P(s)$ by (2).
end for
for Part Q do
 Build G_Q and compute d_Q and l_Q as above.
end for
Obtain a rigid transformation T by solving (4) using *minFunc*, where the likelihood K_P and K_Q can be evaluated using (3).

¹Available from <http://www.di.ens.fr/~mschmidt/Software/minFunc.html>

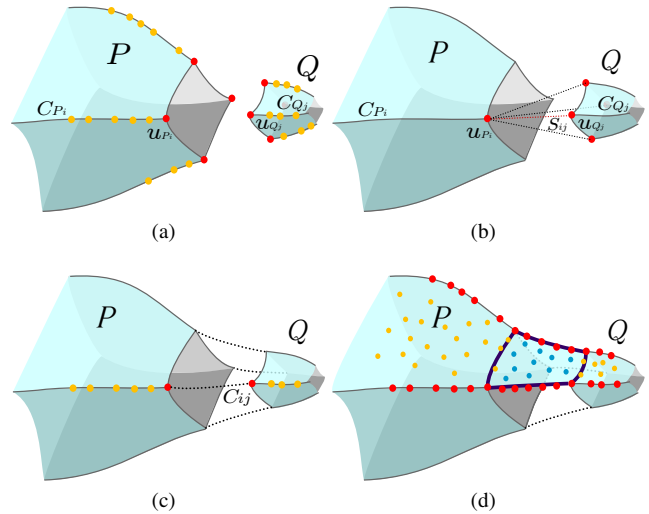


Figure 6: Illustration of the blending process. (a) We first identify seed vertices (shown in red) and then search for more vertices on salient curves (shown in orange). (b) For each seed vertex in one part, we look for the best matching seed vertex in the other part that maximizes the average likelihood on the straight line in between. (c) Vertices on the matching salient curves of both parts are used for a cubic curve fit. (d) The upper patch is constructed by connecting sections of the two open boundaries and two sharp curves (bold lines). The geometry is defined using least-squares meshes.

5 Piecewise Smooth Gap-Filling

Once the two parts are registered, we connect them by filling the gap in between. First, we identify the boundaries and the sharp features which define a number of smooth patches that connect the two parts. Then the patches are parametrized, triangulated and finally instantiated with 3-coordinates [Mallet 1989; Sorkine and Cohen-Or 2004] to form a piecewise smooth transitional surface.

Salient curve interpolation. Given two aligned parts P and Q with salient curves near their boundaries, we first identify sharp edges that have smaller dihedral angles than a threshold Θ (default 65°). For each part we take vertices of those sharp edges adjacent to the open boundary as a set of seed vertices, denoted as $\{u_{P_i}\}$ and $\{u_{Q_j}\}$ for parts P and Q , respectively. We then trace salient curves C_{P_i} and C_{Q_j} from u_{P_i} and u_{Q_j} along consecutive sharp edges; see Figure 6(a). The tracing ends when either a bifurcation is reached or further tracing is not possible. In practice, we found that the above simple procedure is sufficient for handling the models used in our experiments. Nevertheless, more robust techniques, such as [Ohtake et al. 2004; Hildebrandt et al. 2005], can be applied to handle more complex geometries.

After the curve sets $\{C_{P_i}\}$ and $\{C_{Q_j}\}$ are built, they are paired up automatically based on the combined likelihood field $F_{PQ} = F_P + T(F_Q)$. That is, a curve C_{P_i} is matched to C_{Q_j} if the connection between the two has the highest average likelihood; see Figure 6(b). To be precise, the matching curve C_{Q_j} is determined using:

$$C_{Q_j} = \arg \max_j \frac{\int_{S_{ij}} e^{-|F_{PQ}(s)|} ds}{|S_{ij}|}, \quad (5)$$

where S_{ij} is the straight line connecting seeds u_{P_i} and u_{Q_j} with the length $|S_{ij}|$.

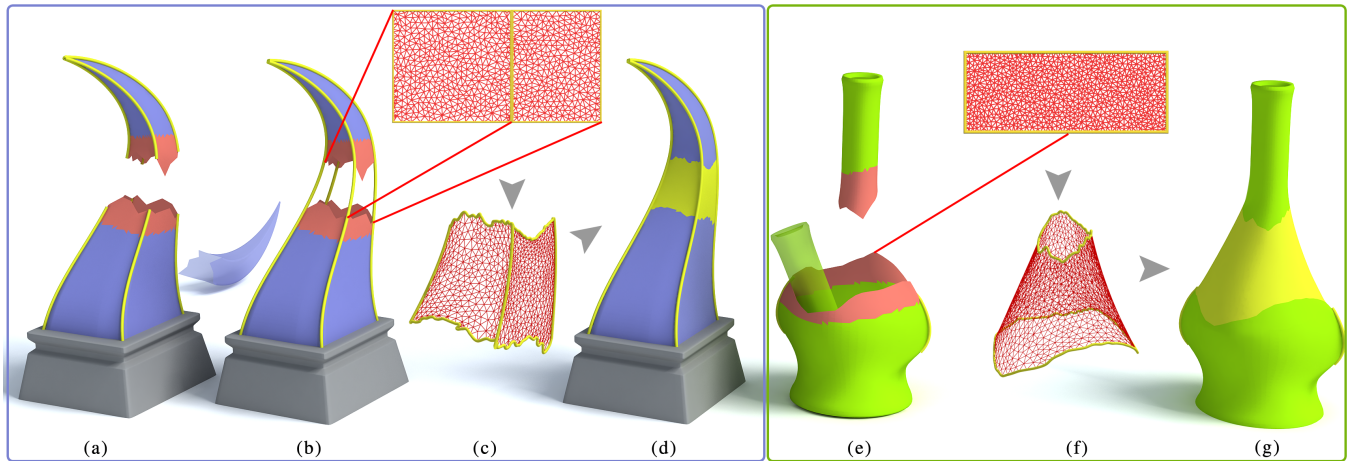


Figure 7: Piecewise smooth gap filling given two registration results shown in (a) and (e). In the horn example on the left, salient curve pairs are interpolated first (b) and then the gaps are filled in a piecewise manner using constrained Delaunay triangulation and least-squares meshes (c). In the bottle example where the top part does not contain any sharp feature, the bridging surface (f) is generated using smooth least-squares meshes directly without the salient curve interpolation step. Both sharp features and smoothness are well preserved in the final compositions (d) and (g).

Two salient curves C_{P_i} and C_{Q_j} are paired up if and only if they mutually match to each other. For each curve pair $\{C_{P_i}, C_{Q_j}\}$, we fit a cubic spline C_{ij} , which is defined as the smoothest curve that fits a set of data points sampled along C_{P_i} and C_{Q_j} ; see Figures 6(c) and 7(b). This automatic pairing method works well on all examples we presented, although additional user interventions might be needed in some tricky situations. It is also worth noting that, when the number of sharp features in the two parts do not match, the above method can still find the most naturally matched pairs, allowing the unmatched salient curves fade out gradually along the transitional surface; see Figures 1A and 8A.

Gap filling. As shown in Figures 6(d) and 7(b), the open boundaries of parts P and Q , together with the two fitted curves, form a *four-sided* gap that needs to be filled. First, we embed the gap into a 2D rectangle that lies on the boundary. Next, apply constrained Delaunay triangulation [Shewchuk 1996] to create new vertices (3D location undefined) and build the connectivity. The geometry of newly introduced vertices is then reconstructed using uniform Laplacian interpolation. Note that we do not have special mesh sampling requirements on input parts P and Q . The positions of vertices in the vicinity of the input parts' open boundaries Ω_P and Ω_Q may be slightly changed for a smoother transition, while the original sampling, regular or irregular, has minimal influence.

After triangulation, we have three sets of vertices: surface vertices $\{I_s\}$ which refer to the original vertices in Ω_P and Ω_Q (yellow in Figure 6(d)), curve vertices $\{I_c\}$ which are the vertices lying on the sharp curves (red in Figure 6(d)), and internal vertices $\{I_f\}$ referring to vertices from the triangulation (blue in Figure 6(d)). We use $I = I_s \cup I_c \cup I_f$ to denote all vertices on the given patch and G to denote vertex connectivity. We set a column vector $\mathbf{x} = [x_1, x_2, \dots, x_i, \dots]_{i \in I}^T$ to hold the coordinates of the vertices in I . The surface vertices and curve vertices (see yellow and red samplings shown in Figure 6(d)) are the control points for the patch, and are denoted by $\{y_i\}_{i \in I_s \cup I_c}$. We compute \mathbf{x} as the minimizer of the linear least squares problem

$$\|L\mathbf{x}\|^2 + \sum_{i \in I_s \cup I_c} w_i^2 |x_i - y_i|^2, \quad (6)$$

where the matrix L is the uniform Laplace operator for the patch that only considers the connectivity G as in [Sorkine and Cohen-Or 2004]. The adaptive weights for surface vertices $\{y_i\}_{i \in I_s}$ are defined based on their distance to the area to be filled. That is, $w_i = e^{-\left(\frac{r_i}{\bar{r}}\right)^2}$, where r_i is the Euclidean distance between y_i and the centroid of the gap, and \bar{r} is the centroid of $\{r_i\}_{i \in I_s}$. Evidently, the closer a surface vertex is to the gap, the larger its deformation may be, allowing for a smooth blending. For curve vertices $\{y_i\}_{i \in I_c}$, we set a high weight and do not update them with $\{x_i\}_{i \in I_c}$ to preserve the connection between patches as well as sharp features; see Figures 7(a-d).

It is worth noting that if either part P or Q has no sharp features detected, we skip the salient curve interpolation step and generate the smooth bridging meshes directly using points on $\Omega(P)$ and $\Omega(Q)$ as control points; see Figure 7(e-g). That is, we simply ignore any sharp features on part P or Q and perform natural mesh interpolation within one patch.

6 Results

Experiments were run on an Intel Core i7 CPU 860@2.80GHz with 2GB RAM. The registration time is roughly proportional to the number of input points within the neighborhood of open boundaries and to the resolution of the discretized FCF. For instance, with 100 original vertices, the average FCF (with a resolution of $100 * 100 * 100$) computation time was around 2 mins and the field-guided alignment time was about 1 min. The average gap filling time for all presented models was less than 10 seconds.

Figures 1 and 8 generally demonstrate the robustness of our composition technique against discrepancies in size, shape, feature and boundary characteristics of the two parts. In Figure 8, the same bottom part of perfume bottle is paired with top parts from different models. While the open boundary of bottom part has a barrel-distorted square shape, the open boundaries of top parts have octagon (8A), pincushion-distorted square (8B), and round shapes (8C & 8D) with complex and asymmetric cuts. Note that the smaller the top part is (like 8D), the further it is displaced away from the bottom part by our registration scheme to allow a natural composition. These kinds of compositions are useful for tasks such as those cov-

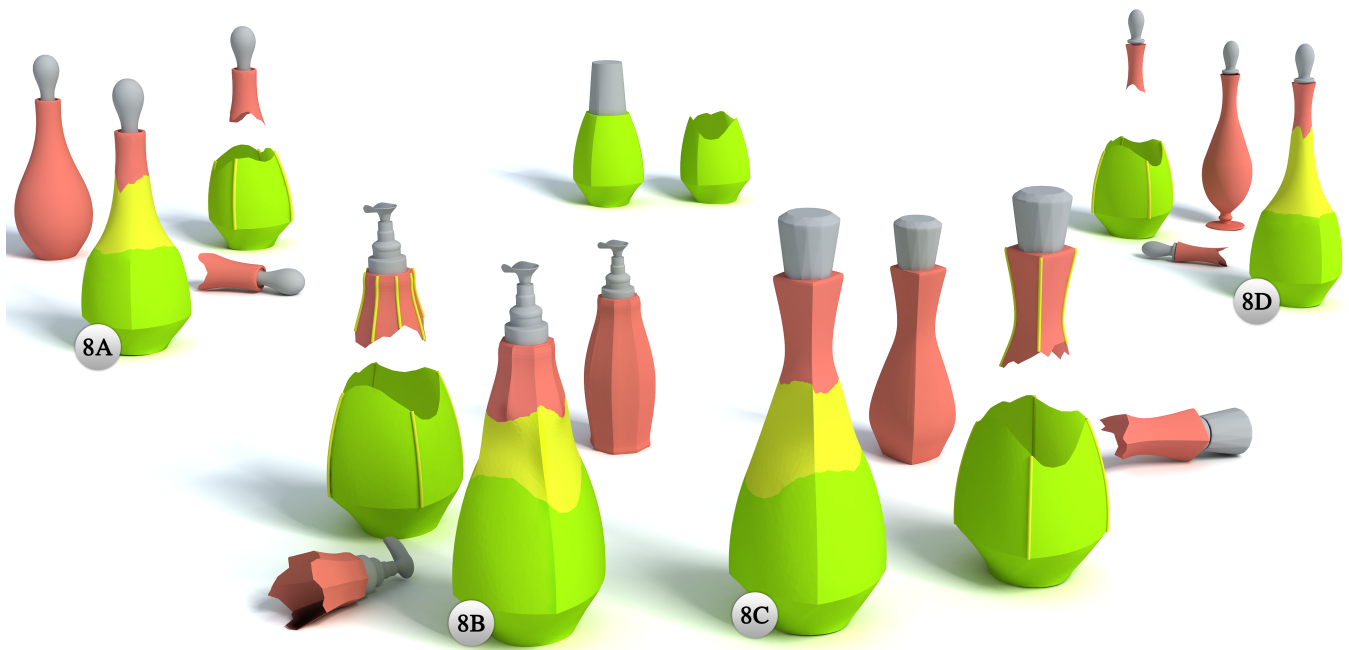


Figure 8: Transplanting of shape parts: broken perfume bottles lying on the floor are aligned and blended with the same bottom part shown in green. All parts used in these examples have non-flat boundaries. The top and bottom parts to be composed may differ in size, shape, and boundary and feature characteristics, necessitating field-guided registration and blending with non-conforming feature profiles. Note in particular 8A with unmatched feature counts and 8C with composition between a piecewise smooth part and a smooth part.



Figure 9: Composition of parts with more than one open boundaries. The parts come from the models on the left; in the middle we present automatic field-guided registration results, where the alignment step searches for a rigid transformation based on multiple pairs of open boundaries; the right side shows two final models obtained after gap filling.

ered in “modeling by example” [Funkhouser et al. 2004].

Results with more complex geometries are shown in Figures 9-12 to demonstrate our algorithm’s ability in the following aspects:

- **Multiple open boundaries (Figure 9).** The optimal registration result needs to take into account the alignments of more than one pair of open boundaries.
- **Partial overlap (Figure 10).** The composed parts partially overlap after optimal field-guided alignment. In order to construct the bridging surface properly, we apply a backward mesh *receding* step, which automatically trims the two registered parts so that the approximated Hausdorff distance between two is beyond a user-specified threshold. Specifically, we check the closest distances from boundary vertices on one part to the other in an alternating fashion. If the distance from a particular vertex is below a threshold (set as doubling the av-

erage edge length of input two meshes by default), we remove it and its adjacent faces. This receding operation iterates until no vertex with distance lower than the threshold exists.

- **Noisy parts (Figure 11).** In general, a thorough piecewise smoothing prior to defining the vector field will certainly work. Here, to further test the noise resistance of our field-guided registration, we only applied a bilateral filtering [Öztireli et al. 2009] on input normals before computing the FCF, and the following registration behaves rather robustly, leading to feature-conforming compositions.
- **Mismatched features (Figure 12).** In the case where the numbers of salient feature curves on the two parts do not match, we use equation (5) to select the best matched curve pairs and let the unmatched ones blend into smooth regions. Note that our field-guided registration still provide proper alignments between the parts.

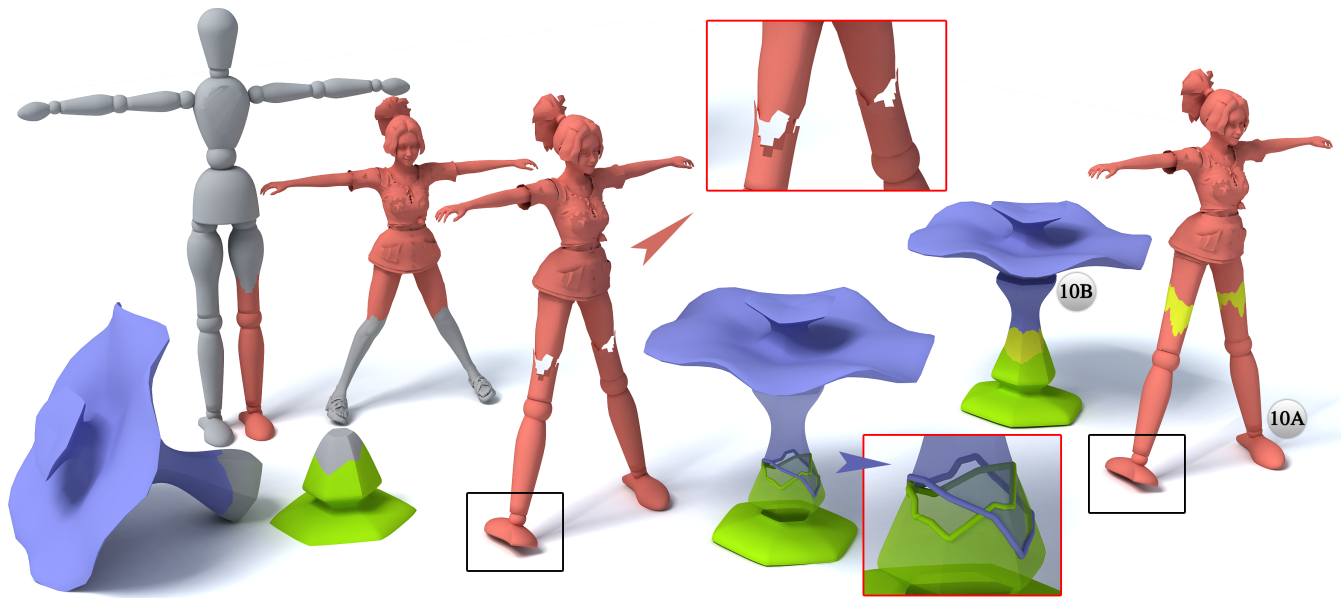


Figure 10: Handling of partial overlap. The optimal alignment found by our registration overlaps the two parts; see zoom-ins. In these cases, the bridging surfaces are constructed by applying an automatic backward mesh receding step before gap filling. Note that one foot (marked by black boxes) is facing the wrong direction since the registration is based on geometric match around the open boundaries.



Figure 11: Robustness to noise in the composed parts. Pre-smoothing via bilateral filtering is sufficiently robust for our registration to behave well. Note that the model 11B also serves as an example with multiple boundaries.

Enforcing alignment constraints. Although our field-guided registration is designed to work fully automatically, the method itself has the flexibility of incorporating additional alignment constraints given by the user. This is desirable since our method is purely geometry-based without considering model semantics. With user assistance, more semantically meaningful compositions can be obtained. For example, without any constraint, the composition shown in Figure 11A (also in Figure 13B) has the human body facing the side of the horse, instead of the front of it, since it yields a more geometrically coherent blending between the two models. To obtain a proper Centaur model, the user can manually specify the front directions for both models. Our registration then searches for the best alignment under the constraint that the two front directions align with each other. This gives us a transformation matrix that

is optimal in the reduced search space and provides a more desired composition; see Figure 13A. Note that even though we only show the use of a direction alignment constraint here, other position- and direction-based constraints can also be enforced.

7 Discussion and Future Work

We have presented a technique to compose a new shape by connecting existing parts together. The technique automatically registers two parts by aligning and blending them. The key is that we do not require the parts to conform to each other, nor to have an overlap. This considerably loosens the requirements on the two connected parts; they can then be cut from their original shapes only roughly, simplifying the shape modeling and composition process. Further-

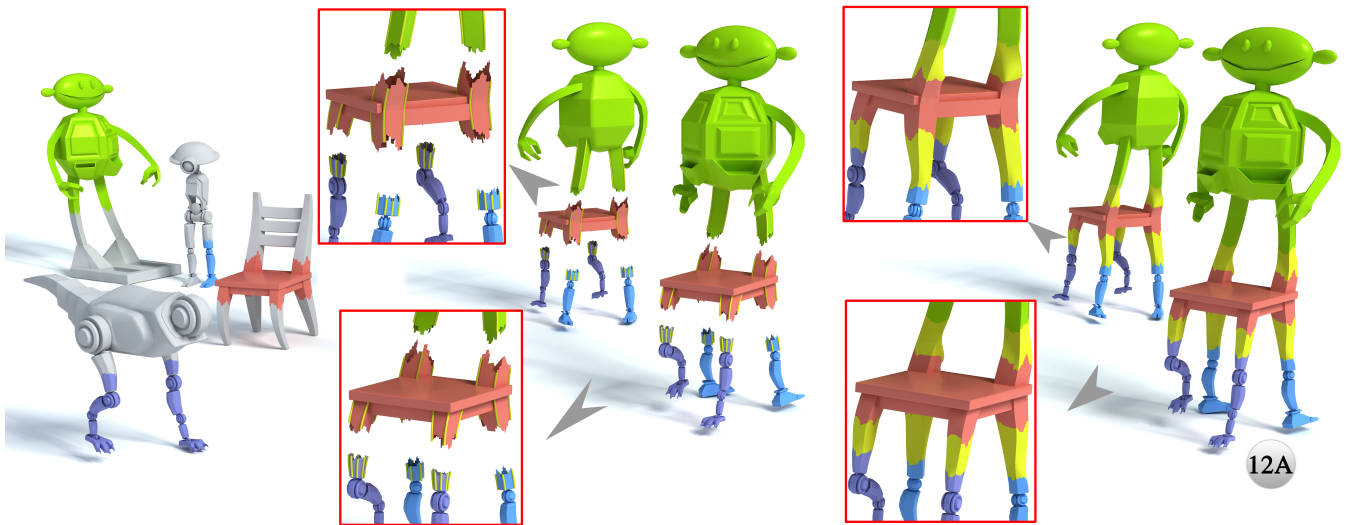


Figure 12: Composition with unmatched feature counts. The chair has four-sided open boundaries, whereas the legs of the green, blue, and purple robots have three-, six-, and eight-sided ones, respectively. The zoom-ins reveal how the bridging surfaces connect different boundaries and blend feature curves (highlighted with yellow) into smooth regions. Note however that our field-guided registration still works effectively.

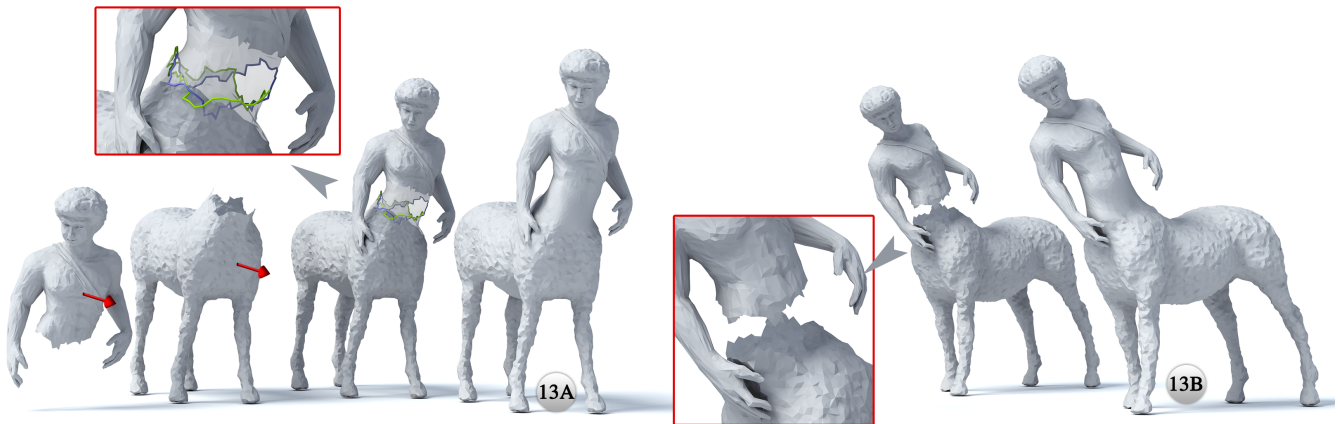


Figure 13: Centaur model obtained with and without user specified constraints. Through enforcing the front directions of the input models (shown in red arrows) to align with each other, our method produces a proper centaur model 13A. In contrast, the model obtained without any constraint, e.g., 11A that has been repeated here as 13B for a better side-to-side comparison, is geometrically more coherent but semantically less desirable.

more, our method is geared toward man-made models, focusing on shapes with sharp features. As we show through examples in this paper, the parts are composed together while respecting their sharp features, forming a composition using piecewise smooth surfaces.

The key problem that the technique faces is therefore how to automatically align non-conforming parts, while possibly leaving gaps in between to allow the construction of piecewise-smooth bridging surfaces. Toward this end, we introduce a field-guided registration mechanism that aligns the two connecting parts through ambient field continuation and surface-to-field registration. To enable sharp feature preservation, our feature-conforming field extrapolates both normals and positions of input surfaces into the ambient space. The experimental results show that our technique can automatically and naturally compose together parts, regardless whether or not sharp features are present.

Limitations and future work. Our current method is limited to registering only two parts at a time, and both need to have their open boundaries well-defined. How to automatically align three or more pieces simultaneously is an interesting question that requires further investigation. In addition, we now consider the geometric match only in the vicinity of the open boundaries of two parts, without taking into account the whole shapes and their correspondence. Hence, while our registration results provide natural interpolations between input parts, they may not be semantically desirable compositions; see Figure 10A and Figure 11A(13B) for example. The enforcement of additional alignment constraints may be needed and could be given by the user in these cases. Finally, while our registration method is capable of handling complicate boundary cuts, the combination of drastic change of normals and highly unbalanced feature distributions along boundaries may lead to unnatural alignment results; see Figure 14.

In the future, besides addressing the above limitations, resolving

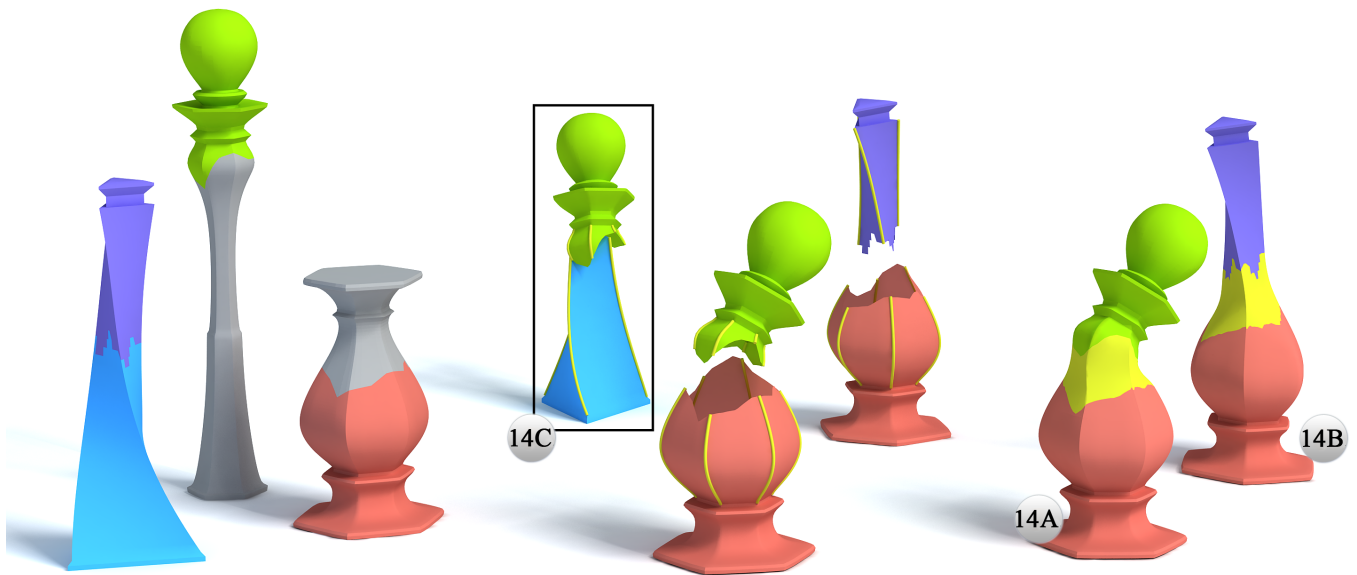


Figure 14: Our method may yield unnatural alignments when the surface normals change drastically near open boundaries and the feature distribution is unbalanced due to asymmetric cuts.

topological differences and generally the handling of complicated topologies of the parts or boundaries will be our focus. The field-guided approach for dealing with complex topologies seems to be a good starting point. We also like to investigate the possibility of adopting our method for the restoration of ancient artifacts.

Acknowledgments

The authors would like to thank all the reviewers for their valuable comments. Thanks also go to Mark Schmidt for his Matlab code *minFunc*, which has been modified to solve our unconstrained optimization problem. The models in Figures 11 and 13 are courtesy of the AIM@SHAPE Shape Repository. This work is supported in part by grants from NSFC (61103166 and 61232011), Guangdong Science and Technology Program (2011B050200007), National 863 Program (2011AA010503), Shenzhen Science and Innovation Program (CXB201104220029A), Natural Science and Engineering Research Council of Canada (293127 and 611370) and the Israel Science Foundation.

References

- AIGER, D., MITRA, N. J., AND COHEN-OR, D. 2008. 4-points congruent sets for robust pairwise surface registration. *ACM Trans. on Graph* 27, 3, 85:1–85:10.
- BELTOWSKA, J., MUSETH, K., AND BREEN, D. 2008. Investigations of tensor voting modeling. In *Proc. of Int. Conf. in Central Europe on Comp. Graphics, Vis. and Comp. Vision*.
- BIERMANN, H., MARTIN, I., BERNARDINI, F., AND ZORIN, D. 2002. Cut-and-paste editing of multiresolution surfaces. *ACM Trans. on Graph (Proc. of SIGGRAPH)* 21, 312–321.
- CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. 2001. Reconstruction and representation of 3D objects with radial basis functions. In *Proc. of SIGGRAPH*, 67–76.
- CHANG, W., AND ZWICKER, M. 2009. Range scan registration using reduced deformable models. *Computer Graphics Forum* 28, 2, 447–456.
- CHANG, W., AND ZWICKER, M. 2011. Global registration of dynamic range scans for articulated model reconstruction. *ACM Trans. on Graph* 30, 3, 26:1–26:15.
- CHAUDHURI, S., KALOGERAKIS, E., GUIBAS, L., AND KOLTUN, V. 2011. Probabilistic reasoning for assembly-based 3D modeling. *ACM Trans. on Graph* 30, 35:1–35:10.
- FU, H., LAN TAI, C., AND ZHANG, H. 2004. Topology-free cut-and-paste editing over meshes. In *Proc. of Geo. Modeling and Processing*, 173–182.
- FUNKHOUSER, T., KAZHDAN, M., SHILANE, P., MIN, P., KIEFER, W., TAL, A., RUSINKIEWICZ, S., AND DOBKIN, D. 2004. Modeling by example. *ACM Trans. on Graph* 23, 3, 652–663.
- FUNKHOUSER, T., SHIN, H., TOLER-FRANKLIN, C., CASTAÑEDA, A. G., BROWN, B., DOBKIN, D., RUSINKIEWICZ, S., AND WEYRICH, T. 2011. Learning how to match fresco fragments. *J. Comput. Cult. Herit.* 4, 7:1–7:13.
- GAL, R., AND COHEN-OR, D. 2006. Salient geometric features for partial shape matching and similarity. *ACM Trans. on Graph* 25, 130–150.
- GELFAND, N., MITRA, N., GUIBAS, L., AND POTTMANN, H. 2005. Robust global registration. In *Symp. on Geom. Proc.*, 197–206.
- HASSNER, T., ZELNIK-MANOR, L., LEIFMAN, G., AND BASRI, R. 2005. Minimal-cut model composition. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, 72–81.
- HILDEBRANDT, K., POLTHIER, K., AND WARDETZKY, M. 2005. Smooth feature lines on surface meshes. In *Symp. on Geom. Proc.*, 85–90.

- HUANG, Q., FLÖRY, S., GELFAND, N., HOFER, M., AND POTTMANN, H. 2006. Reassembling fractured objects by geometric matching. *ACM Trans. on Graph* 25, 569–578.
- HUANG, X., FU, H., AU, O. K.-C., AND TAI, C.-L. 2007. Optimal boundaries for Poisson mesh merging. In *Proc. of ACM Symposium on Sol. and Phys. Modeling*, 35–40.
- ITSKOVICH, A., AND TAL, A. 2011. Surface partial matching and application to archaeology. *Comput. Graph.* 35, 2, 334–341.
- JONES, M. W., BÆRENTZEN, J. A., AND SRAMEK, M. 2006. 3D distance fields: A survey of techniques and applications. *IEEE Trans. Vis. & Comp. Graphics* 12, 581–599.
- KALOGERAKIS, E., CHAUDHURI, S., KOLLER, D., AND KOLTUN, V. 2012. A probabilistic model for component-based shape synthesis. *ACM Trans. on Graph (Proc. of SIGGRAPH)* 31, 4.
- KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Symp. on Geom. Proc.*, 61–70.
- KREAVOY, V., JULIUS, D., AND SHEFFER, A. 2007. Model composition from interchangeable components. In *Proc. of Pacif. Conf. on Comp. Graphics and Applications*, 129–138.
- LÉVY, B. 2003. Dual domain extrapolation. *ACM Trans. on Graph (Proc. of SIGGRAPH)* 22, 364–369.
- LI, X., AND GUSKOV, I. 2005. Multi-scale features for approximate alignment of point-based surfaces. In *Symp. on Geom. Proc.*, 217–226.
- LIN, J., JIN, X., WANG, C., AND HUI, K.-C. 2008. Mesh composition on models with arbitrary boundary topology. *IEEE Trans. Vis. & Comp. Graphics* 14, 653–665.
- MALLET, J. L. 1989. Discrete smooth interpolation. *ACM Trans. on Graph* 8, 121–144.
- MEDIONI, G., AND KANG, S. B. 2004. *Emerging topics in computer vision*. Prentice Hall.
- MUSETH, K., BREEN, D. E., WHITAKER, R. T., AND BARR, A. H. 2002. Level set surface editing operators. In *Proc. of SIGGRAPH*, 330–338.
- NOCEDAL, J., AND WRIGHT, S. J. 2006. *Numerical Optimization (2nd ed.)*. New York: Springer-Verlag.
- OHTAKE, Y., BELYAEV, A., AND SEIDEL, H.-P. 2004. Ridge-valley lines on meshes via implicit surface fitting. *ACM Trans. on Graph (Proc. of SIGGRAPH)* 23, 609–612.
- ÖZTIRELI, C., GUENNEBAUD, G., AND GROSS, M. 2009. Feature preserving point set surfaces based on non-linear kernel regression. *Comp. Graphics Forum* 28, 2, 493–501.
- RASTRIGIN, L. A. 1963. The convergence of the random search method in the extremal control of a many parameter system. *Automation and Remote Control* 24, 1337–1342.
- RUSINKIEWICZ, S., AND LEVOY, M. 2001. Efficient variants of the ICP algorithm. In *Proc. of Int. Conf. on 3D Digital Imaging and Modeling*, 145–152.
- SCHMIDT, R., AND SINGH, K. 2010. MeshMixer: an interface for rapid mesh composition. In *ACM SIGGRAPH 2010 Talks*, 6:1–6:1.
- SHARF, A., BLUMENKRANTS, M., SHAMIR, A., AND COHEN-OR, D. 2006. SnapPaste: an interactive technique for easy mesh composition. *The Visual Computer* 22, 835–844.
- SHARF, A., LEWINER, T., SHKLARSKI, G., TOLEDO, S., AND COHEN-OR, D. 2007. Interactive topology-aware surface reconstruction. *ACM Trans. on Graph* 26, 3, 43:1–43:10.
- SHEWCHUK, J. R. 1996. Triangle: engineering a 2D quality mesh generator and Delaunay triangulator. In *First ACM workshop on Applied Computational Geom.: Towards Geometric Engineering*, vol. 1148 of *Lecture Notes in Comp. Sci.*, 203–222.
- SINGH, K., AND PARENT, R. E. 2001. Joining polyhedral objects using implicitly defined surfaces. *The Visual Computer* 17, 7, 415–428.
- SORKINE, O., AND COHEN-OR, D. 2004. Least-squares meshes. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, 191–199.
- SORKINE, O., COHEN-OR, D., LIPMAN, Y., ALEXA, M., RÖSSL, C., AND SEIDEL, H.-P. 2004. Laplacian surface editing. In *Symp. on Geom. Proc.*, 175–184.
- TAKAYAMA, K., SCHMIDT, R., SINGH, K., IGARASHI, T., BOUBEKEUR, T., AND SORKINE, O. 2011. GeoBrush: interactive mesh geometry cloning. *Computer Graphics Forum (Proc. of Eurographics)* 30, 2, 613–622.
- TANG, C.-K., AND MEDIONI, G. 2002. Curvature-augmented tensor voting for shape inference from noisy 3D data. *IEEE Trans. Pat. Ana. & Mach. Int.* 24, 858–864.
- TOLER FRANKLIN, C., BROWN, B., WEYRICH, T., FUNKHOUSER, T., AND RUSINKIEWICZ, S. 2010. Multi-feature matching of fresco fragments. *ACM Trans. on Graph (Proc. of SIGGRAPH Asia)* 29, 6, 185:1–185:12.
- VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2011. A survey on shape correspondence. *Computer Graphics Forum* 30, 6, 1681–1707.
- XU, K., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2012. Fit and diverse: Set evolution for inspiring 3D shape galleries. *ACM Trans. on Graph (Proc. of SIGGRAPH)* 31, 4, 57:1–57:10.
- YU, Y., ZHOU, K., XU, D., SHI, X., BAO, H., GUO, B., AND SHUM, H.-Y. 2004. Mesh editing with Poisson-based gradient field manipulation. *ACM Trans. on Graph* 23, 3, 644–651.

