Morfit: Interactive Surface Reconstruction from Incomplete Point Clouds with Curve-Driven Topology and Geometry Control

Kangxue Yin[1]   Hui Huang[1]*   Hao Zhang[2]   Minglun Gong[3]   Daniel Cohen-Or[4]   Baoquan Chen[5]

[1]Shenzhen VisuCA Key Lab / SIAT   [2]Simon Fraser University

[3]Memorial University of Newfoundland   [4]Tel-Aviv University   [5]Shandong University

Abstract

With significant data missing in a point scan, reconstructing a complete surface with sufficient geometric and topological fidelity is highly challenging. We present an interactive technique for surface reconstruction from incomplete and sparse scans of 3D objects possessing sharp features. A fundamental premise of our interaction paradigm is that directly editing data in 3D is not only counterintuitive but also ineffective, while working with 1D entities (i.e., curves) is a lot more manageable. To this end, we factor 3D editing into two "orthogonal" interactions acting on skeletal and profile curves of the underlying shape, controlling its topology and geometric features, respectively. For surface completion, we introduce a novel skeleton-driven *morph-to-fit*, or *morfit*, scheme which reconstructs the shape as an ensemble of generalized cylinders. Morfit is a hybrid operator which optimally interpolates between adjacent curve profiles (the "morph") and snaps the surface to input points (the "fit"). The interactive reconstruction iterates between user edits and morfit to converge to a desired final surface. We demonstrate various interactive reconstructions from point scans with sharp features and significant missing data.

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

Keywords: Interactive surface reconstruction, incomplete point cloud, morph and fit, sweep, topology and geometry control

Links: ◆DL 🗎PDF 🗊WEB ◉VIDEO 📁DATA ⬇CODE

1 Introduction

Geometric modeling and reconstruction from laser scan data remains a fundamental and challenging problem in computer graphics. With rapid advances in scanning technology, 3D scanners are producing point clouds at ever increasing resolution and accuracy. However, obtaining a scan that completely covers a complex 3D shape is still difficult, if not impossible, due to physical limitations such as poor surface reflectance or inaccessibility by the scanner. Hence, reconstructing a complete surface with sufficient geometric and topological fidelity, amid significant amount of missing data, is arguably the key challenge in surface reconstruction [Schnabel et al. 2009; Shalom et al. 2010; Harary et al. 2014].

---

*Corresponding author: Hui Huang (hhzhiyan@gmail.com)

Figure 1: *A sparse and incomplete point cloud (on the left) acquired by scanning a shiny metal artwork (in the inset) is interactively reconstructed (in the middle) by our morfit technique with sweeping red profile curves (geometry control) along its green skeleton (topology control).*

Surface reconstruction over missing data is an ill-posed problem. Without any prior, it is even impossible to tell whether a region with missing data is the result of under-scanning or a hole in the shape. A common and effective prior would be the assumption that the scanned object is connected and watertight. However, filling the wrong holes obviously leads to topological error. When the missing data becomes significant, automatic topology inference can hardly be reliable [Sharf et al. 2007a]. State-of-the-art methods for surface reconstruction often employ surface smoothness as the geometry prior [Carr et al. 2001; Davis et al. 2002; Kazhdan et al. 2006], which does not account for possible sharp features in the input shape; see Figure 2. Example-based completion [Sharf et al. 2004; Pauly et al. 2005; Harary et al. 2014] is typically designed to fill few holes in an otherwise complete or accurately reconstructed shape from which the employed exemplars can be extracted.

In this paper, we focus on surface reconstruction from incomplete and sparse scans of 3D shapes possessing sharp features. The problem setting poses challenges to existing priors for topology or geometry inference, necessitating user guidance to obtain desired and accurate reconstruction. An important premise of our interaction paradigm is that directly editing surfaces or point clouds in 3D is not only counterintuitive but also ineffective, while working with 1D entities (i.e., curves) is a lot more manageable [Wu et al. 2012; Chen et al. 2013]. A 3D shape can often be well represented by two sets of "orthogonal" curves: skeletons and cross-sectional pro-
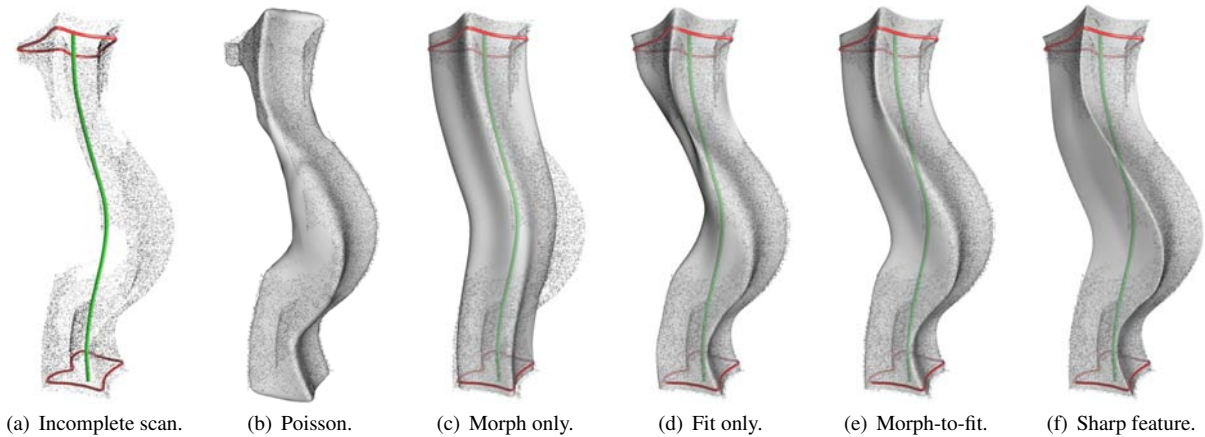
| (a) Incomplete scan. | (b) Poisson. | (c) Morph only. | (d) Fit only. | (e) Morph-to-fit. | (f) Sharp feature. |

Figure 2: *Given a raw scan with significant missing data (a), morfit-based surface reconstruction (e) leads to superior results compared to classic Poisson [Kazhdan et al. 2006] reconstruction (b), as well as to morph-only (c) and fit-(and smoothing)-only (d) surface completion. Morfit also enables recovery and even enhancement of sharp features (f).*

files. It defines a general shape space as combinations of generalized cylinders. As such, we factor 3D editing into two types of independent interactions acting on the two sets of curves:

- In one, the user edits an extracted curve skeleton of the input point scan to exert topology control over the reconstruction process. Also, curves can be sketched over the latent surface to indicate locations of sharp features. One may view such interactions as *sweeping along* the input shape.

- In the second, more fine-grained geometry control can be provided by user editing of *cross-sectional* profile curves, along the curve skeleton or feature curves. These profile curves are seen as orthogonal to the sweep directions above.

The main technical contribution is a novel *morph-to-fit*, or *morfit*, scheme for skeleton-driven surface reconstruction. Morfit is performed between point-sampled cross-sectional shape profiles along a sweep across a skeletal branch. The branch surface reconstructed interpolates (the "morph") between adjacent curve profiles and snaps (the "fit") to the input point cloud, by optimizing an objective function incorporating a data fitting term and a morphing term. The interactive reconstruction iterates between user edits and morfit to converge to a desired final surface.

Unlike previous approaches, e.g., [Sharf et al. 2004; Pauly et al. 2005; Harary et al. 2014], morfit completes surfaces over missing data without explicitly identifying holes or distinguishing between holes and sparse data. Treating the curve skeleton as time axis, morfitting between cross-sectional slices has certain resemblance to time-varying registration [Chang et al. 2012]. However, it differs from previous attempts in several ways: (i) it does not only register, but also integrates correspondence with shape completion; (ii) it does not rely on any pre-set template since no single template can be applicable to all profiles; (iii) it extends the classical global smoothness prior to a piecewise smooth one and respects any detected or specified sharp features; see Figures 1 and 2.

We demonstrate interactive surface reconstruction via curve editing and morfitting on a variety of point scans with significant missing data. In such cases, no fully automatic scheme should be expected to obtain faithful reconstruction results. On the other hand, user guidance is proven to be effective and essential in topology inference, handling of complex joints between surface branches, as well as recovery of geometry details such as sharp features.

2 Related work

Point cloud data for surface reconstruction are typically acquired by a laser scanner. The raw point samples are usually unorganized, unoriented, and incomplete. Due to self-occlusion, less than ideal lighting or inappropriate material of the model, missing parts are prevalent in scan data. The literature on surface reconstruction is extensive and in this section, we mainly discuss methods which deal with missing data.

Surface reconstruction and completion. Common reconstruction techniques estimate an implicit function from an input scan, which smoothly interpolates over regions of missing data [Hoppe et al. 1992; Carr et al. 2001; Alexa et al. 2001; Kazhdan et al. 2006]. Diffusion-based hole filling [Davis et al. 2002] is simple and efficient, but limited for small missing parts. Example-based surface completion [Sharf et al. 2004; Pauly et al. 2005; Harary et al. 2014] is capable of handling large holes, but requires explicit hole detection and proper exemplars from auxiliary data or complete reconstruction in other parts of the input. All of these works focus on geometry completion via smooth interpolation or detail transfer. Wu et al. [2012] introduce a schematic surface reconstruction for architectural scenes from sparse 3D point clouds, where the scene representation is concisely composed of a network of planar transport and profile curves. Our approach deals with more general shape topology and geometry, allowing piecewise smooth surface reconstruction with feature preservation/enhancement and without explicit hole detection.

Topology inference and editing. The key to topology inference during surface reconstruction is the determination of inside/outside with respect to the underlying shape. Huang et al. [2009] improve normal estimation for more reliable separation between close-by surface sheets. Shalom et al. [2010] adopt a weak visibility prior to refine the estimation of signed distances near regions of significant missing data. Theoretically speaking, topological guarantees require sufficiently high sampling density [Amenta and Bern 1998]. With sparse data over regions of close-by surfaces, user interaction becomes necessary [Sharf et al. 2007b]. We edit the shape topology by manipulating curve skeletons. Similar interaction paradigms have been employed for topology editing of polygonal mesh models [Ju et al. 2007; Zhou et al. 2007].
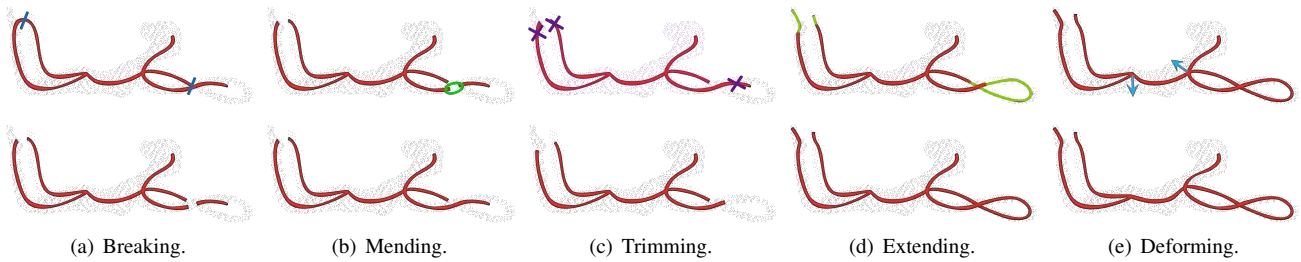
|(a) Breaking.|(b) Mending.|(c) Trimming.|(d) Extending.|(e) Deforming.|

Figure 3: *Skeleton editing, with top row showing user strokes and bottom row the corresponding effects. Given an initial curve skeleton with topological errors, "breaking" (a) is applied at two spots to disconnect skeletal branches that are incorrectly connected. "Mending" (b) is then used to reconnect selected branches based on the correct topology. To adjust the length of existing branches, the user can either apply a "trimming" (c) or perform an "extending" operation along a user specified path (d). Finally, the positions of skeletal branches and joints can be adjusted through the "deforming" operation (e), leading to the final topologically correct skeleton.*

The work of Sharf et al. [2007b] is generally regarded as the state of the art on interactive topology inference for surface reconstruction. However, their work is not designed to handle large amounts of missing data or to recover sharp shape features. The interactive paradigm is quite specific: it allows the user to draw scribbles on a 2D tablet to locally mark inside/outside, where automatic topology recovery is deemed unreliable. The effectiveness of the user interaction critically depends on the proper placement of the tablets. With significant missing data, automatic suggestions for tablet placement can hardly be reliable. In contrast, our method allows versatile editing, not just inference, of the geometry and structure of the scanned shape with fine-grained topology control, enabling the handling of large amounts of missing data.

Curve skeletons and surface reconstruction. A few methods [Sharf et al. 2007a; Tagliasacchi et al. 2009; Li et al. 2010; Huang et al. 2013] have been developed for curve skeleton extraction from incomplete scans and skeleton-driven surface reconstruction. Sharf et al. [2007a] and Li et al. [2010] evolve deformable snakes, relying on surface tension control or smoothness priors for automatic topology and geometry reconstruction. Tagliasacchi et al. [2009] and Huang et al. [2013] both compute medial curve skeletons and rely on smooth cross-sectional curve interpolation along the skeletons for surface completion. None of the above approaches is designed to deal with large amounts of missing data (where user guidance is critical) or shapes with sharp features. Tagliasacchi et al. [2011] reconstruct non-cylindrical geometry by extending curve skeletons with medial sheets. Surface completion over missing data takes a volumetric approach but still relies on the smoothness prior and does not account for sharp features.

Dynamic registration and 3-sweep. Morfitting between cross-sectional slices along a curve skeleton shares similarities to time-varying registration for deformation shape reconstruction [Chang et al. 2012]. In our problem setting, we do not only register curve profiles from different slices, but also perform morphing and geometry completion. Geometry completion over missing data is implicitly achieved by using a template, e.g., [Li et al. 2009], which serves as the geometric prior for reconstruction. Our morfit scheme does not rely on a fixed template, but takes advantage of user edits on the curve profiles for faithful geometry reconstruction, including sharp features. Finally, the sweep motion of our morfit scheme bears some resemblance to 3-sweep [Chen et al. 2013], which employs strong symmetry and smoothness priors to fit image data. Our work operates on 3D point cloud data and deals with much richer varieties of shape topology and curve profiles.

3 Overview of morfit

Our algorithm treats a shape to be reconstructed as one consisting of generalized cylinders centered around curve skeletons. Each branch of the curve skeleton is associated with a portion of the scanned point cloud having the shape of a generalized cylinder; see Figure 2 for an example. We introduce a novel geometric operation called morfit to reconstruct these cylinders. As the name suggests, a morfit is a combined operator that morphs and fits. It interpolates (morphs) between adjacent flat, cross-sectional profile curves and at the same time fits the interpolated surface to the point cloud.

Figure 2 illustrates the morfit concept. The basic setting is depicted in Figure 2(a) where a branch of the curve skeleton (in green) is centered along a sparse and incomplete point cloud; two control profile curves (in red) are defined at two ends of the branch. Poisson reconstruction, shown in Figure 2(b), does not work well due to the significant amount of missing data. A pure data fitting or morph interpolation may be defined to complete the missing parts, only considering or completely disregarding the point cloud, respectively. Clearly, neither the data-only (Figure 2(d)) nor the data-oblivious (Figure 2(c)) reconstruction is acceptable. Fusing the morph with a fit produces the result shown in Figure 2(e).

Surface reconstruction via morfit is the result of optimizing the reconstructed surface by an objective function that includes (i) a morphing term, which enforces the surface to resemble the two controls curves, and (ii) a fitting term, which constrains the surface to lie close to points in the input scan. A notable unique feature of the morfit operator is that the reconstructed surface can contain sharp features along the surface; see Figure 2(f). The ridge created over the surface is the interpolation of the user-specified sharp features over the cross-sectional profile curves.

4 Skeleton-driven interaction

Given an incomplete point cloud, our reconstruction algorithm starts by extracting an initial curve skeleton using the recent method of Huang et al. [2013] on ℓ_1-medial skeletons. The initial skeleton may contain topological and geometric errors, in which case the user can correct them interactively through skeleton editing.

4.1 Skeleton editing

We develop five intuitive operations to allow the user to refine an inaccurate curve skeleton interactively; see Figure 3. For 3D data, editing existing skeletons using these five simple operations is much more effective than directly creating/drawing them from scratch.

- **Breaking** allows the user to disconnect branches at a joint or between joints at specified locations.

- **Mending** allows two or more disconnected skeletal branches to be connected. Both mending and breaking are useful for editing connectivity between skeletal branches.

- **Trimming** is performed at an endpoint of a skeletal branch to erase a portion of erroneous skeleton.

- **Extending** is also performed at endpoints of existing branches and allows skeletal branches to grow along user-specified paths. While the user only provides 2D input for the path, the system automatically maps it to its 3D counterpart based on nearby point cloud data.

- **Deforming** is also a geometry editing operation like "extending". It allows the user to drag a skeletal point so as to displace a nearby portion of the skeleton with moving least squares [Schaefer et al. 2006].

Figure 3 shows a possible skeleton editing scenario. Using a combination of the above five operations, users can easily correct topological errors and generate proper skeletons for the input point cloud, as we show in the accompanying video. It is worth noting that different editing sequences can be used to achieve the same goal.

4.2 Skeleton partition

Provided with a correct skeletal representation of the input point cloud, as far as the user is concerned, the next step is to decompose the skeleton into multiple, potentially overlapping or intersecting, branch paths. This effectively partitions the input point cloud into a few generalized cylinders, each of which is associated with a branch path and will be processed through morfit (Section 5). For shapes with simple skeleton topologies, our algorithm can automatically decompose the skeleton into branch paths by combining branches that give the smoothest connection at each joint. For more complicated structures, as the one shown in Figure 1, user assistance is required, where the user scribbles along the skeleton to indicate how different branches can be joined together to form a branch path.

Next, the whole skeleton is uniformly sampled into a number of skeletal points $\{p_i\}$, with the spacing that is defaulted to 2% of the diagonal length of the input bounding box. For each point p_i, a plane that is perpendicular to the skeleton curve at p_i forms a cross-sectional slice s_i along the skeleton. Each original point q from the input point cloud Q is associated with its closest skeletal point p. As a result, the set of raw input point samples is decomposed into multiple subsets, each of which is associated with a branch path and will be approximated by a generalized cylinder. Note that a given raw point q may belong to multiple subsets since different branch paths may overlap and share portions of their branches.

5 Morfit surface reconstruction

For a given raw scan, the above mentioned process provides us a curve skeleton, which is decomposed into multiple branch paths. User intervention ensures that the skeleton is topologically correct. We can now reconstruct the shape by treating each surface branch associated with a branch path as a generalized cylinder. In the following, we first elaborate how our scheme morfit or sweep along a single branch path, and then describe shape feature enhancing as well as cylinder merging for a quality surface reconstruction.
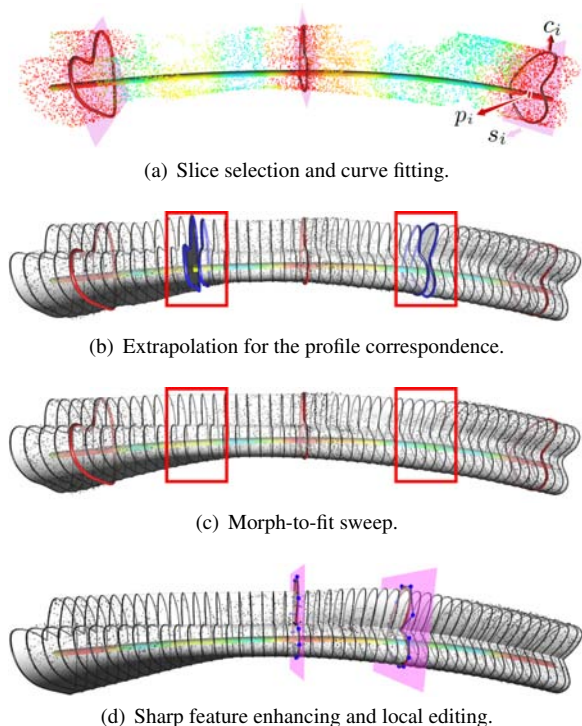

(a) Slice selection and curve fitting.

(b) Extrapolation for the profile correspondence.

(c) Morph-to-fit sweep.

(d) Sharp feature enhancing and local editing.

Figure 4: *Sweep with morfit: (a) based on the completeness values calculated for 2D point clouds on different slices, three locally most complete slices are selected for fitting profile curves; (b) extrapolating each profile curve along the skeletal branch until they meet so that the correspondence among NURBS control points can be established; (c) a smooth surface is then reconstructed by applying a morph-to-fit (morfit) sweep; and (d) it can be easily sharpened or edited interactively while respecting scan data.*

5.1 Profile generation

Given a branch path, the sweep process starts by collecting all raw input points associated with sampled skeletal points $\{p_i\}_{0 \leq i \leq N}$ along the path; this set of points serve as input to next morfit. These raw points are projected onto their corresponding cross-sectional slices, where correspondence is based on the smallest projected distances, forming a 2D point cloud on each slice s_i. Due to noise and incompleteness of the raw scan, instead of fitting a profile curve at each slice, we only perform fitting on a sparse set of slices with high confidence. These slices are selected based on the *completeness* of their associated 2D point clouds.

The completeness of the 2D point cloud on a given slice s_i is estimated by an *angular gap*. To do so, we set up a 2D coordinate system on s_i, whose origin is located at the skeletal point p_i. Next, for each point in the 2D point cloud, we form a unit vector pointing from the origin to that point. We radially sort these unit vectors, such that each 2D unit vector has two neighboring vectors, one on each side. The completeness ω_i is inversely proportional to the maximum angle between any pair of neighboring vectors. By definition, slices that cut through large holes on the input raw scan will have lower completeness values. Figure 4(a) visualizes the completeness score ω_i calculated for different slices s_i.

After the completeness values are evaluated, a small number of non-adjacent slices, shown as pink transparent planes in Figure 4(a), which have the largest values within local neighborhoods are se-
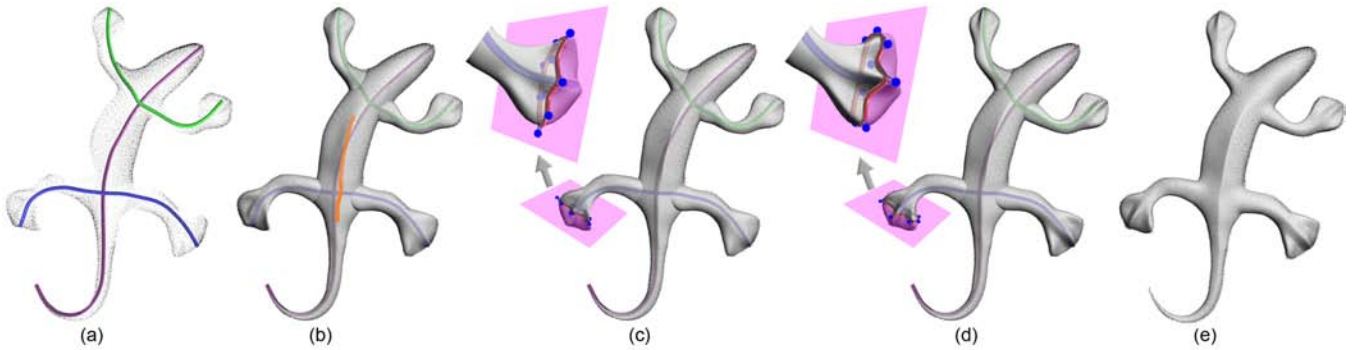
Figure 5: *Enhancing sharp features and merging: given the skeleton for the gecko shape (a), three branch paths (purple, blue, and green) are automatically detected, leading to the reconstruction of three generalized cylinders (b); sharp features can be enhanced by interactive scribbling, e.g., orange stroke on the surface (b), or editing a profile curve on a cross-sectional slice (c-d); merging the enhanced cylinders gives us a manifold surface (e) and smoothing is performed at the intersections between the cylinders.*
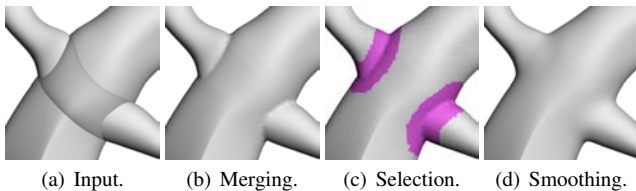

| (a) Input. | (b) Merging. | (c) Selection. | (d) Smoothing. |

Figure 6: *A zoomed-in view on the gecko model in Figure 5, showing how CSG operations are used to automatically merge intersecting cylinders (b). If necessary, local smoothing (d) around the merged area (c) can be applied as well.*

lected for profile curve generation. Note that here we pick the most, but not necessarily fully, complete slices, so the method works even when all the slices are incomplete. NURBS, which is often applied in reserve engineering [Piegl and Tiller 2001], is used here for fitting a profile curve c_i to a given 2D point cloud, since it allows gradient discontinuity that is useful for modeling sharp features. In practice, we apply the NURBS fitting method from [Wang et al. 2006]. The fitted NURBS is represented using a set of 2D control points, $\{c_i^k\}$, in the 2D coordinate system defined on slice s_i. We maintain the same number (10 by default) of control points for all NURBS fitting profile curves over the curve skeleton.

5.2 Morfit sweep

With profile curves generated on the selected slices, our next objective is to automatically compute a curve for every slice by morphing existing profile curves and fitting the input data at the same time. This is performed in two stages. The first stage extrapolates each existing profile curve to its neighboring slices until the curves propagated from different profile curves meet at a slice in-between. This allows matching the control points among profile curves. The second stage optimizes all curves along the branch that both interpolate the profiles and fit the input data. After the two stages, the surface is reconstructed and represented as a sequence of 2D curves.

5.2.1 Extrapolation

During the first stage, we extrapolate each profile curve outward to its neighboring profile curves with two objectives: i) the deformation of the curve shall be *as-rigid-as-possible* [Igarashi et al. 2005] and ii) the deformed curve shall fit the raw input data. That is,

assuming a profile curve c_i is already defined for slice s_i, the deformed curve c_j for the neighboring slice s_j ($j = i+1$ or $j = i-1$) is computed by optimizing:

$$\arg \min_{c_j} (E_d(c_j) + \alpha E_m(c_i, c_j)), \qquad (1)$$

where the data fitting term E_d measures the error of using the deformed curve c_j to fit the raw data, and the morphing term E_m keeps the deformation of the curve as-rigid-as-possible. In practice, we define the data fitting term as:

$$E_d(c_j) = \sum_{q \in s_j} d(q, c_j), \qquad (2)$$

where q is a raw point in slice s_j, and d is the function that measures the closest distance between a point and a NURBS curve.

Since both curves c_i and c_j are defined using 2D control points in their respective coordinates, the deformation from c_i to c_j can be represented as a set of 2D offsets $c_j^k - c_i^k$, one for each control point k. To deform a profile curve in an as-rigid-as-possible fashion, we decompose the offsets into a global rigid translation $T_{i \to j}$ and local elastic per-control-point deformations $\delta_{i \to j}^k$. That is, we have $c_j^k = T_{i \to j} c_i^k + \delta_{i \to j}^k$ for all k. Hence, the morph term enforces the shape of the profile curve by penalizing the second term:

$$E_m(c_i, c_j) = \sum_k \left\| \delta_{i \to j}^k \right\|_2. \qquad (3)$$

The equation (3) measures exactly the elastic differences between control points after the rigid transformation (local elastic per-control-point deformation from ith profile to jth profile).

Once the curves extrapolated from two profiles c_i and $c_{i'}$ meet at their intermediate location, we have two NURBS curves defined for the same slice. The shapes of these two curves should be similar since they fit the same 2D point cloud; see Figure 4(b). Thus, the correspondence between those two adjacent blue curves derived from different profiles can be automatically determined by a greedy search. The best match is the one with the minimum summation of Euclidean distances between the corresponding control points. Then the matching relationship is propagated back to the corresponding profiles in order to establish the correspondence relationships among all profiles. These are then used during the interpolation of the profile curves in the second stage.
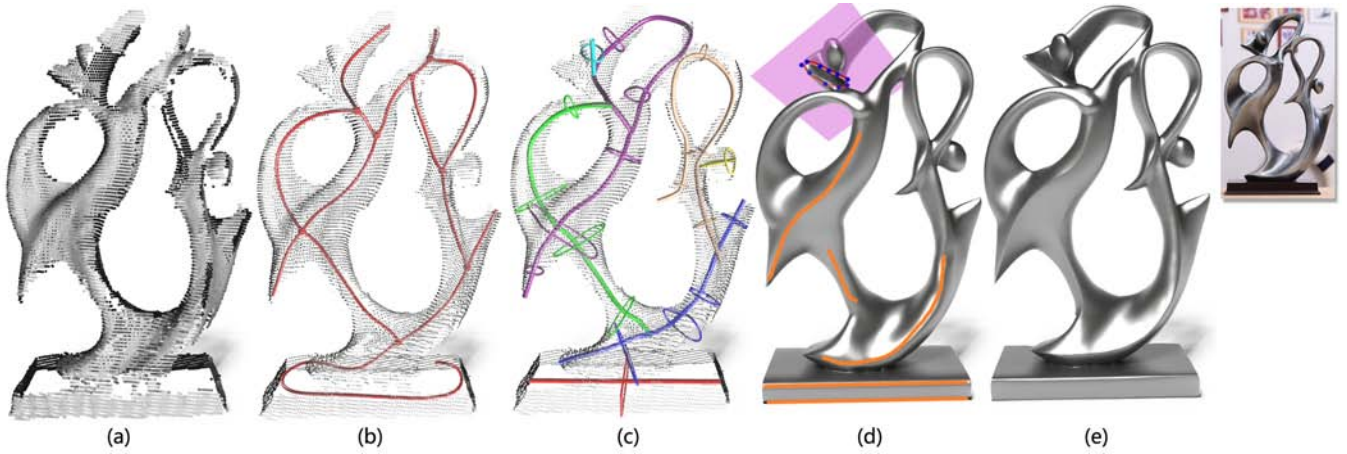
Figure 7: *Given an incomplete scan (a) of a metal sculpture (inset at the right), the automatically generated skeleton (b) contains some topological errors. These errors are corrected interactively, where the revised skeleton is decomposed into several branch paths shown using different colors (c). By applying morfit to each branch path and then performing cylinder merging, the sculpture shape is well reconstructed (d). With more user effort, e.g., indicating where to enhance sharp features, shown as orange lines in (d), and subsequent user editing of a few profile curves, the final reconstructed surface (e) can faithfully model the original sculpture.*

5.2.2 Interpolation

The interpolation stage optimizes all profile curves in a batch processing manner. The objective function consists of three terms: the aforementioned data fitting and morphing terms for keeping the reconstructed curve close to the scanned data and for maintaining the shapes of profiles, and an additional smoothness term to enforce a smooth sweep of the profile curves along the direction of the skeleton. Hence, we optimize the following function:

$$\arg\min_{\{c_i\}} \sum_i \left(E_d(c_i) + \alpha E_m(c_i, c_{i+1}) + \beta E_s(c_{i-1}, c_i, c_{i+1}) \right),$$
(4)

under the constraint that $c_i = c_p$ for all selected profiles $\{c_p\}$. The weights α and β in the above function are empirically set to 0.1 and 0.01, respectively. The smoothness term E_s is defined as:

$$E_s(c_{i-1}, c_i, c_{i+1}) = \sum_k (1 + \cos(\angle(c_{i-1}^k - c_i^k, c_{i+1}^k - c_i^k))), \quad (5)$$

where large angles between two successive slices are penalized during the sweep; compare Figures 4(b) and 4(c). It is worth noting that here we only enforce the smoothness of the reconstructed surface along the skeleton direction. Sharp features dictated by the shapes of the profile curves can be well preserved; see Figures 4(c) and 5.

To compute the optimal $\{c_i\}$, we utilize the Method of Moving Asymptotes (MMA) [Svanberg 1987] and solve equations (1) and (4) using the NLOPT library. MMA applies a special type of convex approximation and has been proven in the past to be an efficient tool for solving structural optimization problems. For each step of the iterative process, a strictly convex approximating sub-problem is generated and solved. The generation of these sub-problems is controlled by the so-called moving asymptotes, which both stabilize and speed up the convergence of the general process. Like many other non-linear optimization methods, a good initial guess is important for the MMA's accuracy and rate of convergence. For the first extrapolation problem (1), we simply copy each profile curve outward its neighboring profile curves as the initial guess. The solution to (1) then naturally serves as the initial guess for solving the objective function (4) in the second interpolation stage.

By default, the above sweep operation results in a surface with the shape of a generalized cylinder, i.e., both ends of the cylinder are open. To close up the cylinder, we check the 2D point clouds on the slices for the two skeletal endpoints. If the maximum distance between any two raw points is larger than a given threshold, we simply attach a planar patch at this end of the cylinder. Otherwise, we know that the cylinder reduces to a tip at this end since all scanned raw points are inside a small area. Hence, we place a degenerated NURBS curve, whose control points share the same location, at the centroid of the 2D point cloud. Including this degenerated curve into the morfit optimization allows the profile curve to be reduced gradually towards the tip point in a plausible manner; see the tail of the gecko in Figure 5.

5.3 Feature enhancement

Due to noisiness and incompleteness of the input raw scan, as well as adoption of the smoothness term, the above automatic profile curve fitting and morfit sweep process may not always be able to accurately model sharp surface features or other geometric details in a shape. To further enhance the reconstructed model with these desired features, we provide two interactive operators.

Sharpening strokes. A *sharpening stroke* is a scribble the user draws on the reconstructed surface to indicate the location of a sharp feature. All curves which intersect the scribble are refitted with higher weights on the closest control points instead of using the default uniform weights for all control points. The high weights are smoothly decayed into normal values outside the scribble, allowing the sharp feature to gradually fade out. This then serves as a good initial guess for another sweep using (4); see comparisons between Figures 5(b) and 5(c), as well as Figures 7(d) and 7(e).

Curve editing. The second operator allows the user to select and edit an individual curve c_i. When the user clicks close to a skeletal point, the corresponding 2D point cloud and control points for the current NURBS fitting curve are displayed. The user can then manipulate the control points directly; see the accompanying video. The edited curve is added into the selected, high-confidence, profile set $\{c_p\}$, to serve as an additional constraint when solving (4);
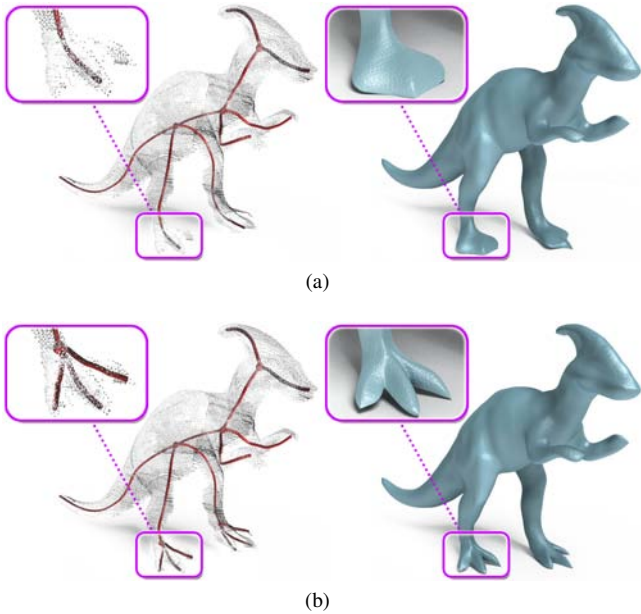
(a)

(b)

Figure 8: *Effect of skeleton editing: ℓ_1-medial skeleton (a) automatically generated from the incomplete point cloud does not capture the fine structures, resulting in lack of details in the reconstructed surface. Adding the toe branches (b) interactively allows us to model the original foot shapes of the dinosaur properly.*

compare Figures 5(c) and 5(d), 7(d) and 7(e). Note that for each edited curve, we have its corresponding slice completeness measure ω_i and can use it to weigh the optimization constraint. Thus, the more incomplete the data is, the more freedom the user may have to change the nearby surface; otherwise, the reconstruction respects the scan. In the local editing result in Figure 4(d), the pink slice to the left is with high completeness and the right one is not.

5.4 Cylinder merging

With sweep performed along all branches, the input scan is represented using a connected set of generalized cylinders; see Figure 6(a) for an intersecting example. Merging of the cylinders is obtained by performing a CSG operation [Rocchini et al. 2001], resulting in a complete 2-manifold (Figure 6(b)). If needed, local regions near an intersection between two cylinders can be identified as in Figure 6(c) and then smoothed; see Figure 6(d). Figures 5(d) and 5(e) show the reconstructed gecko surface before and after the cylinder merging operation in Figure 6.

6 Results

In this section, we show comprehensive reconstruction results obtained by our interactive technique on 3D models containing rich features and complex topologies. All input scans are sparse and incomplete, where the captured objects are difficult to capture in high quality using laser scanners due to shiny surfaces or large self-occlusions. As we demonstrate, our curve-driven interactive reconstruction technique leads to successful interpolation of surfaces over regions with significant missing data, sharp features, and general geometric and topological characteristics, producing fully complete 3D models with high fidelity.
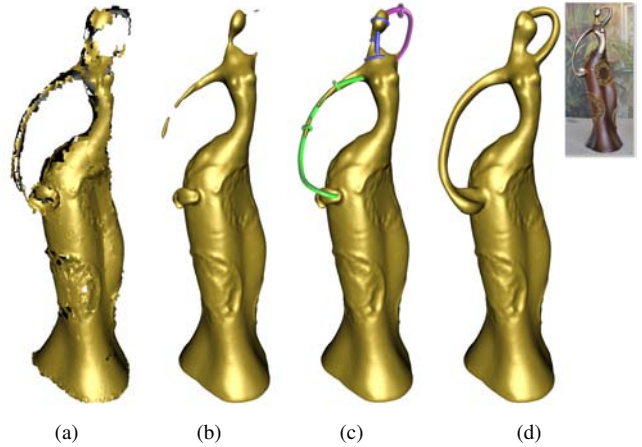

(a) (b) (c) (d)

Figure 9: *Assisting Poisson reconstruction with curve editing. Given an incomplete scan of a real object (a), Poisson reconstruction (b) misses the thin structures. By inserting and editing three curve branches (c), morfit brings the missing structures back (d).*
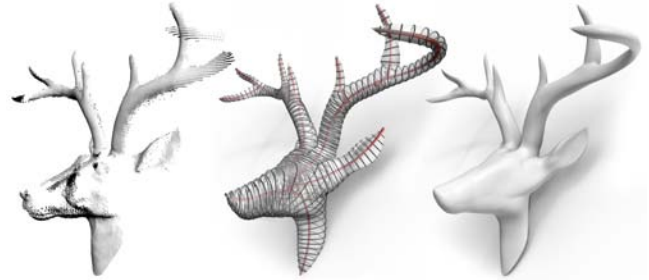

Figure 10: *Given a point cloud with large regions of missing data, our approach properly reconstructs the sharp features. Note that a section of the deer horn is completely missing and can only be precisely reconstructed through user interactions.*

#Fig	1	7	8	9	10	11	12	13	15	16
#BE	6	6	4	0	2	3	0	4	5	7
#CE	3	2	0	0	1	2	3	6	4	4
#PS	5	7	10	3	10	5	4	8	5	12
#SS	3	7	0	0	0	0	0	0	0	4

Table 1: *Number of user edits and other interaction operations performed to generate results shown in various figures throughout the paper. #BE: number of skeletal branches edited/added/removed; #CE: number of profile curves edited; #PS: number of strokes to indicate sweep paths; #SS: number of sharpening strokes.*

Parameters and timing. The default parameter weights $\alpha = 0.1$ and $\beta = 0.01$ in the objective function (4) are applied throughout all the presented experiments except for the examples in Figures 1, 7 and 17(c). There objects have regularly cylindric or rectangular pedestals, hence a larger $\beta = 0.1$ is set to strengthen the smoothness constraint and assist the reconstruction. The morfit sweeping time on an average branch path with 10 control points per NURBS curve is about 1 second. The timing is measured on an Intel Xeon E5-2687W CPU @3.40GHz with 8GB RAM. All the reconstruction results were obtained at interactive speed, as shown in the accompanying video.
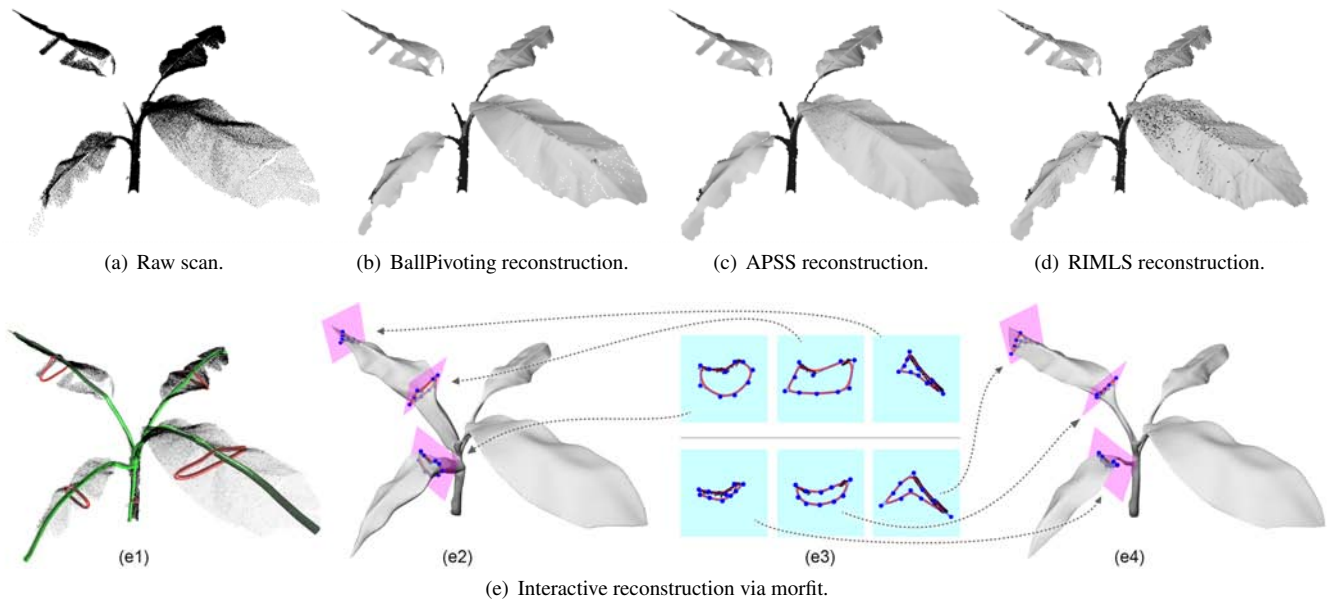
(a) Raw scan.　　(b) BallPivoting reconstruction.　　(c) APSS reconstruction.　　(d) RIMLS reconstruction.

(e1)　　　　　　(e2)　　　　　　(e3)　　　　　　(e4)

(e) Interactive reconstruction via morfit.

Figure 12: *Comparison with automatic surface reconstruction methods including Ball-Pivoting [Bernardini et al. 1999] in (b), APSS [Guennebaud and Gross 2007] in (c), and RIMLS [Öztireli et al. 2009] in (d). Morfit already achieves a better surface completion result (e2) with direct sweep from automatically extracted skeleton (green) and automatically fitted profile curves (red), shown in (e1). However, due to significant missing data, e.g., near a junction between stems, and the dramatic profile change between the leaflet and petiole at the front, the automatic sweep result still contains artifacts. Editing of only three profile curves, as shown in (e3) with top and bottom rows showing the before and after of curve editing, the artifacts are removed, resulting in a faithful organic plant reconstruction (e4).*
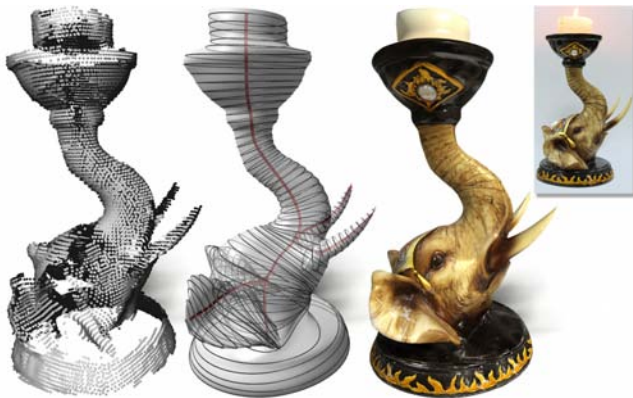

Figure 11: *Reconstruction with texture mapping: given an incomplete scan (left) of a real object (inset), our approach can generate a photo-realistic model (right) after texture mapping that also captures sharp features (e.g., teeth) and thin structures (e.g., ears).*

Figure 13: *The real airplane object (inset) contains non-cylindrical parts and the input point cloud (left) is incomplete. Yet, our approach successfully reconstructs a 3D model (right) that respects the raw scan and is close to the real object. Different branch paths processed by morfit are shown in different colors.*

User interaction. Results from Figures 7 and 8 confirm that missing data and noise in the input scans prevent automatically generated curve skeletons fully capturing the detailed shape structures and can be erroneous. Our method allows a versatile set of edit operations to be applied to address the various needs for structure enhancements and correction of topological errors; again see Figure 7 and the zoom-ins in Figure 8. Such user interactions are not only applicable to raw input scans directly, but can also be employed to enhance the results of existing surface reconstruction schemes. Figure 9 shows how morfit can repair the missing parts resulting from Poisson reconstruction. For a detailed account on how much us-

er effort is required typically, refer to Table 1 and Figure 17, which summarize the number of edit/interaction operations applied to generate all reconstruction results shown in the paper.

Sharp features. Effective recovery of sharp features is illustrated in Figures 1, 7, 10, and also later, in Figures 14, 16 and 17. Note that user scribbles to indicate sharp features do not need to be precise, they only need to be *near* the intended features. Our algorithm is able to snap the scribbles onto high curvature edges that can be detected from input data; it is also able to sharpen the edges or tips as directed. Figure 11 shows a textured reconstruction result with recovered/enhanced sharp features.

Non-cylindrical shapes. While morfit has been designed to effectively handle shapes formed by a combination of generalized cylinders, it is also able to reconstruct shape parts whose profiles are far from being circular-shaped or even star-shaped. For exam-
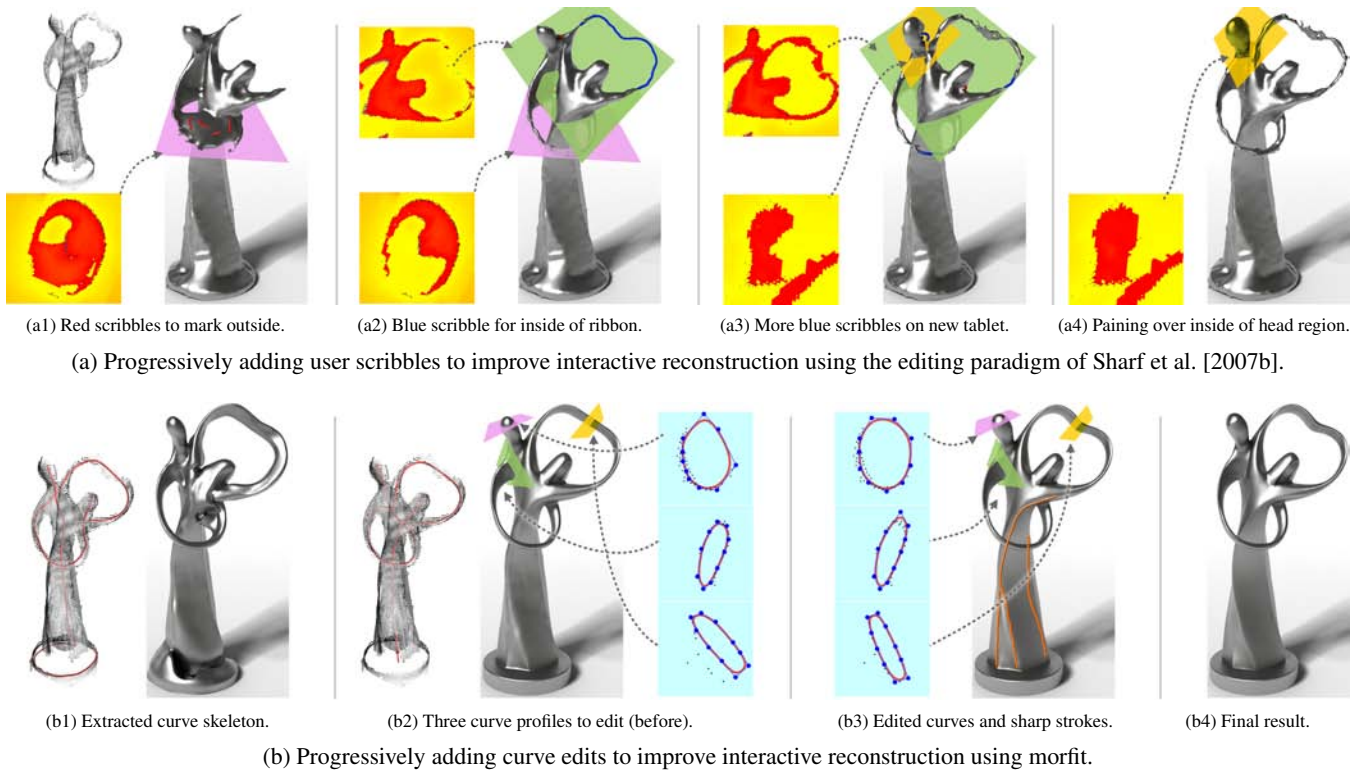
(a1) Red scribbles to mark outside. (a2) Blue scribble for inside of ribbon. (a3) More blue scribbles on new tablet. (a4) Paining over inside of head region.

(a) Progressively adding user scribbles to improve interactive reconstruction using the editing paradigm of Sharf et al. [2007b].

(b1) Extracted curve skeleton. (b2) Three curve profiles to edit (before). (b3) Edited curves and sharp strokes. (b4) Final result.

(b) Progressively adding curve edits to improve interactive reconstruction using morfit.

Figure 14: *Comparing morfit (b) with the state-of-the-art interactive surface reconstruction method by Sharf et al. [2007b] (a) in terms of how user interactions progressively improve the reconstruction. To accurately reconstruct over regions with significant missing data, Sharf et al.'s method requires proper placement of 2D tablets (colored profile planes in top row), on which topology information is specified (red $\rightarrow$ inside, yellow $\rightarrow$ outside). Here, these tablets were placed with great care for achieving better results. However, it is still hard to fully cover regions with significant missing data, e.g., near a thin structure where data is almost entirely missing (a2 and a3). Inside of a tablet, the user needs to correct the wrong topology or geometry by "painting" over wrong inside regions using red scribbles (a1) and wrong outside regions using blue scribbles (a2). This process is quite tedious and prone to imprecisions, e.g., due to slightly misplaced tablets, leading to rough and low-quality reconstruction results (a3 and a4). In contrast, morfit relies on user interaction on 1D curves where 6 skeleton edits, 3 profile curve edits (see before and after shown in (b2) and (b3)), as well as 3 sweep path strokes and 3 sharpening strokes, provide the fine-grained topology and geometry controls enough to obtain a faithful and quality reconstruction (b4).*
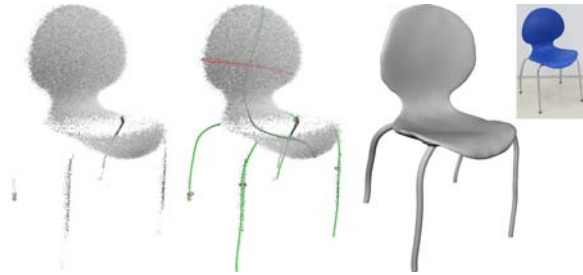

Figure 15: *Reconstruction of highly non-cylindrical parts from highly incomplete scan. Existing skeletonization algorithms cannot handle the challenging data, thus curve editing is necessary. Morfit, performed over hand-created (green) skeletons and five (red) profile curves, leads to a proper reconstruction.*

ple, the animal ears in Figures 10 and 11, the plant leaves in Figure 12, the wings of the airplane in Figure 13, and the chair's back and seat surface in Figure 15 though being thin sheets, can all be properly reconstructed by morfit with a small amount of user effort. Please refer to Table 1 for user interaction counts and the accompanying video for interactive reconstruction sessions.

Comparison to auto reconstruction and [Sharf et al. 2007b].
Figure 12 presents an example which confirms that none of the popular automatic reconstruction methods is able to handle incomplete raw scans beyond a certain degree. With minimal user interactions (four sweep path strokes plus editing on three profile curves), our morfit provides the most faithful reconstruction.

Figure 14 compares morfit with the state-of-the-art interactive reconstruction technique by Sharf et al. [2007b], as both methods work progressively in a reconstruction task with increasing user interaction. Sharf et al.'s method is not designed to handle large amount of missing data. Their topology inference allows the user to mark inside/outside with respect to the underlying surface on a 2D tablet. Properly positioning the tablets for taking best advantage of user inputs is generally difficult. Even when the tablets are best positioned *manually*, as in the cases shown in the top row of Figure 14, it is still tedious to use inside/outside constraints to specify the right geometry and topology when there is significant missing data. The user essentially needs to "paint" over the entire missing data region, as shown, e.g., in (a2) and (a3), where the inside region is a narrow band. The excessive user edits are prone to imprecision, resulting in unsatisfactory interpolation and reconstruction (a4).

In contrast, curve creation and editing provide a more direct means for topological control, as shown in (b1) and (b2). These are combined with refinement of profile curves at sparse locations, as in-
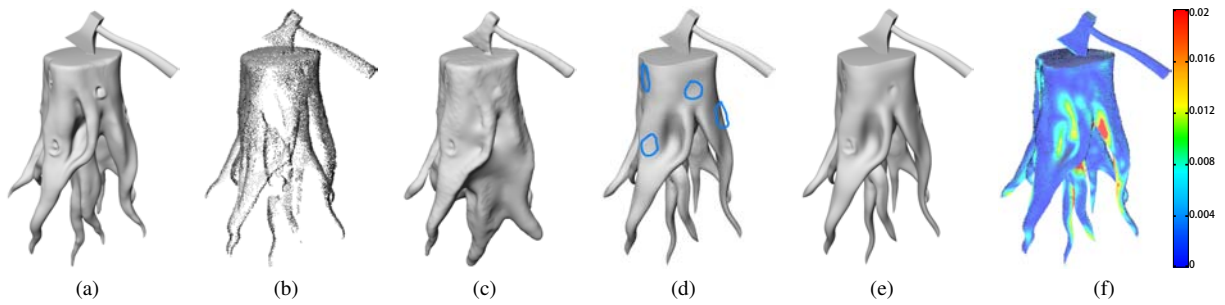
Figure 16: *Evaluating reconstruction accuracy using a synthetic model (a) that contains non-cylindrical parts, sharp features, local geometric details, and much self-occlusion. Given the incomplete virtual scan (b), state-of-the-art (automatic) screened Poisson reconstruction [Kazhdan and Hoppe 2013] gives unsatisfactory result (c). Through a modest amount of user edit (see rightmost column of Table 1), morfit generates a faithful reconstruction (d) of the overall shape. Surface details that were smoothed out during morfit sweeping, e.g., areas highlighted in blue circles in (d), can be enhanced back using local projection approaches, such as RIMLS [Öztireli et al. 2009] in (e). The color plot (f) shows reconstruction errors, measuring closest distance to the original model. The maximum error is less than 0.02 when the virtual model is normalized into a unit cube.*
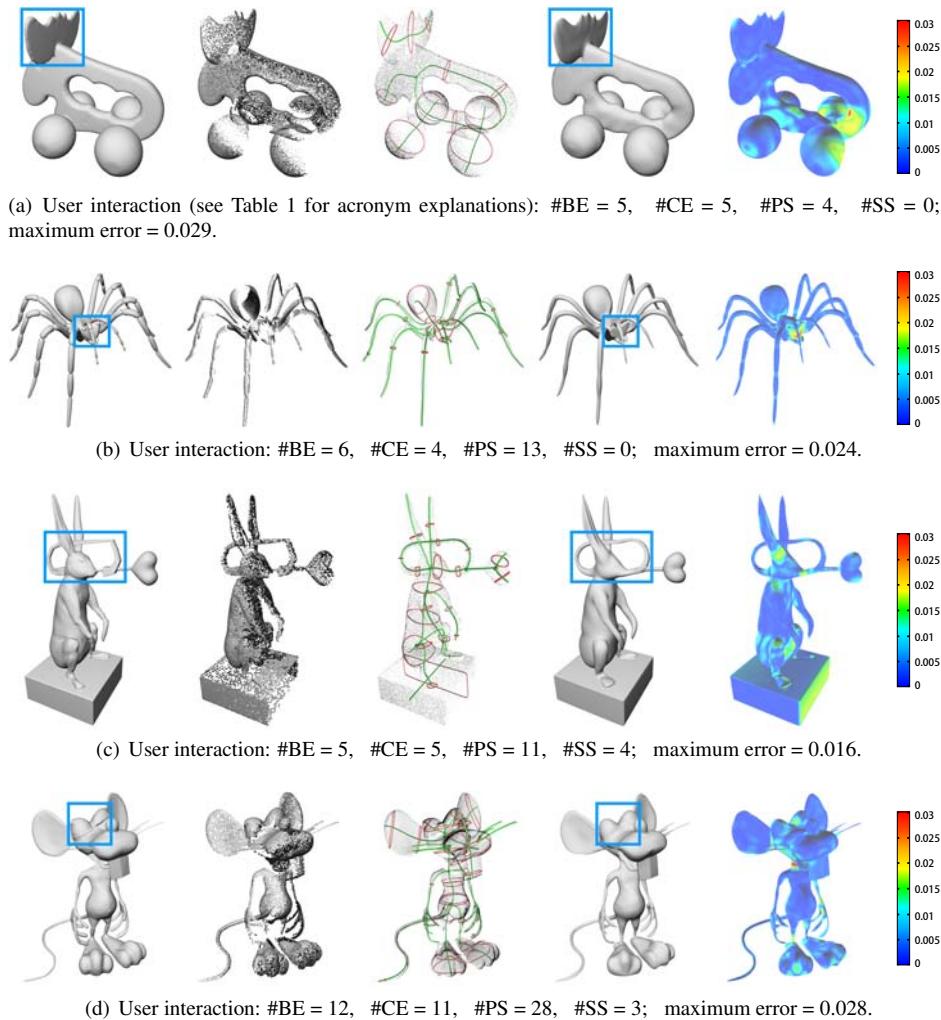

(a) User interaction (see Table 1 for acronym explanations): #BE = 5, #CE = 5, #PS = 4, #SS = 0; maximum error = 0.029.

(b) User interaction: #BE = 6, #CE = 4, #PS = 13, #SS = 0; maximum error = 0.024.

(c) User interaction: #BE = 5, #CE = 5, #PS = 11, #SS = 4; maximum error = 0.016.

(d) User interaction: #BE = 12, #CE = 11, #PS = 28, #SS = 3; maximum error = 0.028.

Figure 17: *Additional morfit reconstruction results with evaluation on four complex models. The columns show the virtual models, virtual scans with significant missing data, curve skeletons (green) and profile curves (red), the final reconstructions, and error plots from left to right. Some surface details were smoothed out, e.g., comparing areas highlighted within the blue boxes, while the overall reconstruction errors (see color plots and the maximum error values in (a)-(d) captions) remain low.*
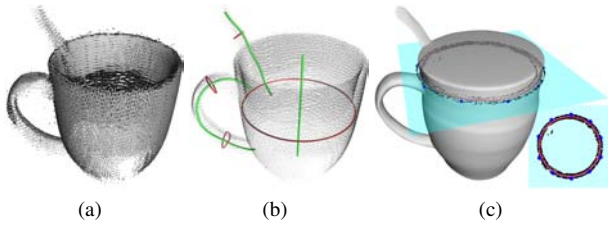
(a)   (b)   (c)

Figure 18: *A shape with two close-by surface layers: our approach merges the two layers into one generalized cylinder and closes the end of the cylinder.*
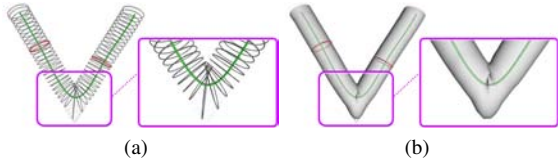

(a)   (b)

Figure 19: *Intersection between profile curves causing reconstruction artifacts. When reconstructing a V-shape with a sharp turn, due to the intersections between profile curves on adjacent slices (a), the reconstructed surface contains "folding" artifacts.*

dicated in (b2) and (b3), leading to a faithful reconstruction (b4) while requiring considerably less user effort. Moreover, [Sharf et al. 2007b] is not designed to recover sharp features.

Reconstruction accuracy. Figure 16 evaluates reconstruction accuracy of morfit using a virtual model that is designed to challenge various aspects of our method. The model consists of branches leading to self-occlusions, sharp edges, as well as non-cylindrical parts. Screened Poisson reconstruction [Kazhdan and Hoppe 2013] in (c) is not satisfactory. With a modest amount of user edit (see Table 1), morfit reconstruction recovers the overall shape (d) faithfully. Surface details present in the input but were smoothed out during the sweeping can then be enhanced back (e) using available local projection techniques such as RIMLS [Öztireli et al. 2009] or texture transfer. To compare the final result with the original virtual model, we color-plot the reconstruction error in (f), where the maximum error is less than 0.02 when the virtual model (a) is normalized into a unit cube, and such errors occur only near areas where there is virtually no scan data, comparing (b) and (f).

Figure 17 further demonstrates the generality of morfit reconstruction on several complex models, where we show the edited 1D curves, report user interaction counts, and display error plots for measuring reconstruction accuracy. The 3D models were downloaded from a dataset for skeletonization [Reniers et al. 2008] and normalized. The virtual scans resulted in significant missing data.

Limitations. Morfit is designed to reconstruct shapes that possess reasonably meaningful curve skeletons. Objects such as a rice bowl or a baseball hat fall somewhat out of scope. Using swept curves for modeling also has its limitations on preserving or recovering high-frequency surface details. Skin transfer, local projection (Figure 16), or texture mapping (Figure 11), can be applied to enhance the final reconstruction. Using morfit to model complicated organic shapes such as plants with dense leaf formations may require much user editing, rendering the technique less practical.

From a technical standpoint, we currently do not deal with generalized cylinders with open boundaries; neither can we easily handle

shapes with multiple close-by layers (Figure 18). In addition, if the skeletal curve contains a sharp turn, the profile curves on adjacent slices may intersect each other, leading to artifacts in the resulting model (Figure 19 for an example). Additional user interventions may be needed to adjust the location of the skeleton curve and/or to edit the profile curves.

7 Discussions and future work

We present an interactive surface reconstruction technique which we call morfit. We demonstrate its performance in producing complete surface meshes from input scans which can be highly incomplete and sparse. Morfit offers the user direct control over the generated geometry and topology, allowing the user to quickly reach a reconstruction at high fidelity. The technique also offers the handling of sharp features under simple user guidance.

There is an obvious trade-off between user effort and quality of surface reconstruction; see Figures 7, 8 and 14. Our intent is for morfit to offer the best trade-off to date. The reconstruction tool we develop is quite general and versatile in that it offers both geometry and topology controls. A carefully designed interaction metaphor only requires the user to perform intuitive manipulation and editing of curves, arguably the simplest yet general geometry primitives to interact with.

In retrospect, interpolation schemes such as RBF and Poisson can be effective over relatively dense scans of highly detailed geometry that contain many small holes. Morfit is not the best at handling missing data in this nature. It excels over regions with large amount of missing data where the geometry is piecewise smooth. We believe that the combination of morphing and data fitting can be further improved to deal with fine surface structures. We would continue investigating how to enhance morfit to handle shapes with delicate details, maintaining both high fidelity and efficiency.

Aside from addressing the aforementioned technical limitations (Section 6), we would also like to focus our effort on addressing two important aspects of surface reconstruction. On one hand, our morfit technique is seemingly moving toward and certainly suggesting an *edit-to-fit* paradigm for surface reconstruction. This would lead to the development of a fully fledged 3D editor with more choices for user interactions. The editor models shapes while closely coupling the editing with the reconstruction of 3D input data, possibly at very low quality. On the other hand, we aspire to further simplify user interactions. This may lead to a surface reconstruction technique that can be prevalently used.

Acknowledgments

References

ALEXA, M., BEHR, J., COHEN-OR, D., FLEISHMAN, S., LEVIN, D., AND SILVA, C. T. 2001. Point set surfaces. In *Proc. Int. Conf. on Visualization*, 21–28.

AMENTA, N., AND BERN, M. W. 1998. Surface reconstruction by voronoi filtering. In *Proc. Symp. on Computational Geometry*, 39–48.

BERNARDINI, F., MITTLEMAN, J., RUSHMEIER, H., SILVA, C., AND TAUBIN, G. 1999. The ball-pivoting algorithm for surface reconstruction. *IEEE Trans. Visualization & Computer Graphics 5*, 4, 349–359.

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. *Proc. of SIGGRAPH*, 67–76.

CHANG, W., LI, H., MITRA, N., PAULY, M., AND WAND, M. 2012. Dynamic geometry processing. In *Eurographics Tutorial*.

CHEN, T., ZHU, Z., SHAMIR, A., HU, S.-M., AND COHEN-OR, D. 2013. 3-sweep: Extracting editable objects from a single photo. *ACM Trans. on Graphics (Proc. of SIGGRAPH Asia) 32*, 6, 195:1–195:10.

DAVIS, J., MARSCHNER, S. R., GARR, M., AND LEVOY, M. 2002. Filling holes in complex surfaces using volumetric diffusion. In *Proc. IEEE Int. Symp. on 3D Data Processing, Visualization and Transmission*, 428–441.

GUENNEBAUD, G., AND GROSS, M. 2007. Algebraic point set surfaces. *Proc. of SIGGRAPH 26*, 3, 23:1–23:9.

HARARY, G., TAL, A., AND GRINSPUN, E. 2014. Context-based coherent surface completion. *ACM Trans. on Graphics 33*, 1, 5:1–5:12.

HOPPE, H., DeROSE, T., DUCHAMP, T., McDONALD, J., AND STUETZLE, W. 1992. Surface reconstruction from unorganized points. In *Proc. of SIGGRAPH*, 71–78.

HUANG, H., LI, D., ZHANG, H., ASCHER, U., AND COHEN-OR, D. 2009. Consolidation of unorganized point clouds for surface reconstruction. *ACM Trans. on Graphics (Proc. of SIGGRAPH Asia) 28*, 5, 176:1–176:7.

HUANG, H., WU, S., COHEN-OR, D., GONG, M., ZHANG, H., LI, G., AND CHEN, B. 2013. ℓ_1-medial skeleton of point cloud. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 32*, 65:1–65:8.

IGARASHI, T., MOSCOVICH, T., AND HUGHES, J. F. 2005. As-rigid-as-possible shape manipulation. *ACM Trans. on Graphics 24*, 3, 1134–1141.

JU, T., ZHOU, Q.-Y., AND HU, S.-M. 2007. Editing the topology of 3D models by sketching. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 26*, 3, 42:1–42:9.

KAZHDAN, M., AND HOPPE, H. 2013. Screened poisson surface reconstruction. *ACM Trans. on Graphics 32*, 3, 29:1–29:13.

KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. *Proc. Eurographics Symp. on Geometry Processing*, 61–70.

LI, H., ADAMS, B., GUIBAS, L. J., AND PAULY, M. 2009. Robust single-view geometry and motion reconstruction. *ACM Trans. on Graphics (Proc. of SIGGRAPH Asia) 28*, 5, 175:1–175:10.

LI, G., LIU, L., ZHENG, H., AND MITRA, N. J. 2010. Analysis, reconstruction and manipulation using arterial snakes. *ACM Trans. on Graphics (Proc. of SIGGRAPH Asia) 29*, 6, 152:1–152:10.

ÖZTIRELI, C., GUENNEBAUD, G., AND GROSS, M. 2009. Feature preserving point set surfaces based on non-linear kernel regression. *Computer Graphics Forum 28*, 2, 493–501.

PAULY, M., MITRA, N. J., GIESEN, J., GROSS, M., AND GUIBAS, L. 2005. Example-based 3D scan completion. *Proc. Eurographics Symp. on Geometry Processing*, 23–32.

PIEGL, L. A., AND TILLER, W. 2001. Parametrization for surface fitting in reverse engineering. *Computer-Aided Design 33*, 8, 593–603.

RENIERS, D., VAN WIJK, J., AND TELEA, A. 2008. Computing multiscale curve and surface skeletons of genus 0 shapes using a global importance measure. *IEEE Trans. Visualization & Computer Graphics 14*, 2, 355–368.

ROCCHINI, C., CIGNONI, P., GANOVELLI, F., MONTANI, C., PINGI, P., AND SCOPIGNO, R. 2001. Marching intersections: an efficient resampling algorithm for surface management. In *Proc. IEEE Int. Conf. on Shape Modeling & Applications*, 296–305.

SCHAEFER, S., McPHAIL, T., AND WARREN, J. 2006. Image deformation using moving least squares. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 25*, 3, 533–540.

SCHNABEL, R., DEGENER, P., AND KLEIN, R. 2009. Completion and reconstruction with primitive shapes. *Computer Graphics Forum (Proc. of Eurographics) 28*, 2, 503–512.

SHALOM, S., SHAMIR, A., ZHANG, H., AND COHEN-OR, D. 2010. Cone carving for surface reconstruction. *ACM Trans. on Graphics (Proc. of SIGGRAPH Asia) 29*, 6, 150:1–150:10.

SHARF, A., ALEXA, M., AND COHEN-OR, D. 2004. Context-based surface completion. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 23*, 3, 878–887.

SHARF, A., LEWINER, T., SHAMIR, A., AND KOBBELT, L. 2007. On-the-fly curve-skeleton computation for 3D shapes. *Computer Graphics Forum 26*, 3, 323–328.

SHARF, A., LEWINER, T., SHKLARSKI, G., TOLEDO, S., AND COHEN-OR, D. 2007. Interactive topology-aware surface reconstruction. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 26*, 3, 43:1–43:10.

SVANBERG, K. 1987. The method of moving asymptotes - a new method for structural optimization. *Int. J. Numerical Methods in Engineering 24*, 2, 359–373.

TAGLIASACCHI, A., ZHANG, H., AND COHEN-OR, D. 2009. Curve skeleton extraction from incomplete point cloud. *ACM Trans. on Graphics (Proc. of SIGGRAPH) 28*, 3, 71:1–71:9.

TAGLIASACCHI, A., OLSON, M., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2011. Vase: Volume-aware surface evolution for surface reconstruction from incomplete point clouds. *Computer Graphics Forum (Proc. Eurographics Symp. on Geometry Processing) 30*, 5, 1563–1571.

WANG, W., POTTMANN, H., AND LIU, Y. 2006. Fitting b-spline curves to point clouds by curvature-based squared distance minimization. *ACM Trans. on Graphics 25*, 2, 214–238.

WU, C., AGARWAL, S., CURLESS, B., AND SEITZ, S. M. 2012. Schematic surface reconstruction. In *Proc. IEEE Conf. on Computer Vision & Pattern Recognition*, 1498–1505.

ZHOU, Q., JU, T., AND HU, S. 2007. Topology repair of solid models using skeletons. *IEEE Trans. Visualization & Computer Graphics 13*, 4, 675–685.