A THINNING ALGORITHM FOR TOPOLOGICALLY CORRECT 3D SURFACE RECONSTRUCTION

Leonid Tcherniavski
Department of Informatics
University of Hamburg
tcherniavski@informatik.uni-hamburg.de

Peer Stelldinger
Department of Informatics
University of Hamburg
stelldinger@informatik.uni-hamburg.de

ABSTRACT

The existing algorithms for 3D surface reconstruction from point clouds require objects with smooth surfaces and extremely dense samplings with no or only very few noise in order to guarantee a correct result. Moreover they can only be applied to single object reconstruction and not to segmentation of three or more regions. We have developed an alternative surface reconstruction method wich preserves the topological structure of multi-region objects under much weaker constraints. It is based on the Delaunay complex and α-shapes and uses a local thinning algorithm for the reconstruction of region boundaries. In this work we give a detailed analysis of its behaviour and we compare it with other approaches.

KEY WORDS

Surface Reconstruction, Topology Preservation, Alpha-Shapes, Delauney Trianguation.

1 Introduction

A fundamental question of image analysis is how closely a computed image segmentation corresponds to the underlying real-world partition. Existing geometric sampling theorems are limited to binary partitions, where the space is split into (not necessarily connected) fore- and background components. In this case, the topology of the partition is preserved under various reconstruction schemes when the original regions are sufficiently smooth and the sampling is dense enough, e.g. see [4, 5]. Some approaches can also deal with noisy samplings, e.g. see [11, 12].

To our knowledge, all existing approaches for topologically correct surface reconstruction need an object to have a smooth surface, i.e. the surface curvature has to be everywhere bounded. This is often given by a bound on the so-called local feature size, which is the minimal distance of a surface point to the nearest point on the medial axis [4, 5, 11], but also by giving global smoothness constraints [12]. This is impossible for surfaces with edges (like e.g. a cube). Moreover they can deal only with binary partitions of the space, i.e. the problem of separating an object from its background by reconstructing the object boundary. Out aim is to be able to separate objects in more complicated configurations, i.e. to segment the 3D space into several regions, each being topologically correctly reconstructed.

2 Topologically Correct Reconstruction

Since edges naturally occur when three or more regions meet, a reconstruction algorithm for multi-object segmentation tasks must be able to deal with non-smooth boundaries. Therefore we introduced the concept of r-stable partitions:

A partition of the space $\mathbb{R}^3$ into several regions r_i is called *r-stable* when its boundary $B = \bigcup \partial r_i$ can be dilated by a closed ball of radius s without changing its homotopy type for any $s \leq r$.

Thus, regions in an r-stable partition do not change their topological information when one thickens the boundary moderately. Note, that r is also known as "homological feature size" [8]. Now, a *sampling* of the boundary is a set of *(sampling) points* which lie near the boundary B:

A finite set of sampling points $S = \{s_i \in \mathbb{R}^3\}$ is called a *(p, q)-sampling of the boundary* B when the distance of every point $b \in B$ to the nearest point in S is at most p, and the distance of every point $s \in S$ to the nearest point in B is at most q.

Thus, p denotes the sampling density, whereas q denotes the samplig accuracy. A sampling with $q = 0$ (i.e. all sampling points lie on the boundary) is called a *strict sampling*. Having a set of sampling points, we can construct the Delaunay complex and the α-complex.

The α-complex $D_\alpha(S)$ of a set of points S is defined as the subcomplex of the Delaunay complex of S which contains all cells C such that (1) the radius of the smallest sphere containing the sampling points of C is smaller than α, and it contains no other point of S, i.e. $C^0 \cap S = \emptyset$, or (2) an incident cell C' with higher dimension is in $D_\alpha(S)$. The polytope $|D_\alpha(S)|$, i.e. the union of all elements of $D_\alpha(S)$, is called *α-shape*.

In case of sufficiently smooth surfaces and dense, noisefree samplings, the α-shape gives a good reconstruction of the original surface without changing topology [4, 5, 15]. But in other cases, the α-shape can not directly be used, it needs to be modified.

Let $D_\alpha(S)$ be the α-complex of a sampling S and $|D_\alpha(S)|$ be its α-shape. Then the components of $|D_\alpha(S)|^c$ are called α-holes of $|D_\alpha(S)|$. The (α, β)-holes of $|D_\alpha(S)|$ are the α-holes H, where the largest radius of some n-cell in H is at least $\beta \geq \alpha$. The union of the α-shape $|D_\alpha|$ with all α-holes of D_α that are not (α, β)-holes is called the (α, β)-*shape reconstruction*.

Filling spurious holes in the α-shape reconstruction is a necessary step for getting a topologically correct boundary reconstruction. But there are also other problems regarding topology: although the (α, β)-shape reconstruction separates the different regions from each other, these regions may have small handles. In order to identify and remove these cases, we apply a homotopy type preserving thinning. We will denote an m-dimensional cell (m-cell) C in a cell complex D as *simple* if the number of cells of D which contain C is equal to one. Now the containing cell must be an $(m+1)$-cell and the removal of the two cells (also called *collapse*, see [9] does neither change the homotopy type of the complex nor the topology of the background regions. Now the thinning algorithm for the (α, β)-shape reconstruction is as follows:

Minimal (α, β)-shape reconstruction algorithm:

1. Given a (p, q)-sampling S of the boundary of some partition of the space, compute the α-complex of S with some $\alpha > p$.
2. Add all cells to the complex, which belong to an α-hole which is no (α, β)-hole for $\beta = \alpha + p + q$.
3. Find all simple m-cells (for any m with $n > m \geq 0$) of the given (α, β)-shape reconstruction and put them in a priority queue by using the radii of the simple m-cells as priority, i.e. m-cells with big radius are the first to be removed.
4. As long as the queue is not empty:
   (a) Get the m-cell e with the highest priority from the queue.
   (b) If e is not simple anymore, it has lost this property during the removal of other cells. Skip the following and recommence with step 4.
   (c) Otherwise, remove e and the adjacent $(m+1)$-cell $t \in \mathcal{R}$ from the boundary reconstruction.
   (d) Check whether the other cells adjacent to t have now become simple and put them in the queue if this is the case.
5. For m going from $n-2$ to 0 do:
   (a) Remove all m-cells of the complex, which do not have an adjacent $(m+1)$-cell in the complex.

In [15] we proved that the minimal (α, β)-shape reconstruction algorithm is able to reconstruct the topological information of multi-object segmenations even in case of relatively coarse and noisy samplings:

Theorem 1 *Let $\mathcal{P}$ be an r-stable partition of the space $\mathbb{R}^n$, and S be a (p, q)-sampling of $\mathcal{P}$'s boundary B. Then the minimal (α, β)-shape reconstruction algorithm results in a cell complex D with $|D|$ having the same homotopy type as B, and the components of B^c are topologically equivalent to the components of $|D|^c$, if (1) $p < \alpha \leq r - q$, (2) $\beta = \alpha + p + q$ and (3) every region r_i contains an open γ-disc with $\gamma \geq \beta + q > 2(p + q)$.*

3 Comparison to other Approaches

Surface reconstruction from sampling points has been of great interest in the last years. An overviev is given in table 1.

One of the first approaches which gives a proof about the correct separation of an object from its background is the crust algorithm[4]. The drawback of this algorithm is, that it results in a triangle set, which only contains a correct triangulation of the surface instead of being one. This was corrected in the cocone algorithm. Both algorithms require relatively dense samplings in order to guaranty their correctness: while the crust algorithm requires a $(p, 0)$-sampling with p being at most equal to 0.1 times the local feature size (lfs), cocone needs $p < 0.06lfs$. In order to have a more intuitive measure we do not directly compare the necessary sampling density p, but the number N of sampling points one needs in order to reach such a density in case of a unit sphere (in this case, the local feature size is 1 and we have simply $p \leq 0.1$). To compute this number is commonly known as the sphere covering problem and there exist very tight lower bounds for estimating this minimal number of sampling points for a given p, e.g. following [13] we get $N \geq 2 + 2\pi / \left(6 \operatorname{arccot}\left(\frac{\sqrt{3}}{2}(2 - p^2)\right) - \pi\right)$. In case of the crust and the cocone algorithm it follows that $N \geq 484$, respectively $N \geq 1344$. Both algorithms have been enhanced in several steps. E.g. power crust [6] is faster than crust and guarantees to reconstruct a correct surface in contrast to just containing one, and the modified power crust algorithm [11] can even deal with $q \leq 0.1$. All three variants require $p \leq 0.1$. Analogously tight cocone is a variant of cocone guaranteeing to reconstruct a watertight surface for the same bound as cocone, $p < 0.06$. In contrast to these approaches much better bounds are given in [12]. Niyogi et al. use a global smoothness criterion instead of a local one, but are able to give much better bounds. In case of $q = 0$ only $p < 0.48$ is needed which is equivalent to only 22 well distributed sampling points, while $p, q < 0.17157$ (at least 165 sampling points) still allows to resonstruct a correct surface. In [15] an even better bound is given: by using the ball-pivoting algorithm the bounds given in [12] can be improved for $q = 0$ to $p < r$, which implies to use only 6 sampling points. As shown in the last section, a homeomorphic surface can also be reconstructed in case of highly noisy samplings, i.e. $p + q < 1$. Note, that in case of a sphere the restriction given by γ requires $p + q < 0.5$, but that in general the samplingdensity can locally be lower since the γ-bound must only be fulfilled somewhere in a region.

All these methods require smooth surfaces. In [7] a method is given, which reconstructs “Lipschitz Surfaces”, and thus allows certain nonsmooth surfaces, but corners cannot have acute angles. The method works with locally adaptive but noise-free samplings and requires more than 263 points to result in a surface which is isotopic to a unit sphere boundary. To conclude, our approach gives better bounds on the necessary sampling density, on the acceptable amount of noise and on the generality of the type of surfaces than previous approaches, and allowes for the first time to regard non-manifold partitions.

	sampling	(p,q)	# of points
Crust [4]	locally adaptive	$p \leq 0.1$ $q = 0$	484
Co-Cone [5]	locally adaptive	$p < 0.06$ $q = 0$	1344
Power Crust [6] Amenta et al. 2000	locally adaptive	$p \leq 0.1$ $q = 0$	484
Tight Co-Cone [10]	locally adaptive	$p < 0.06$ $q = 0$	1344
[12]	global	$p < 0.48$ $q = 0$	22
[12]	global	$p < 0.17157$ $q < 0.17157r$	165
Modified Power Crust [11]	locally adaptive	$p \leq 0.1r$ $q \leq 0.1r$	484
Lipschitz Surfaces [7]	locally adaptive	$p \leq 0.13574r$ $q \leq 0$	263
[15]	global	$p < r$ $q = 0$	6
[14]	global	$p < 0.5r$ $q < r - p$	78

Table 1. Comparison of different surface reconstruction algorithms.

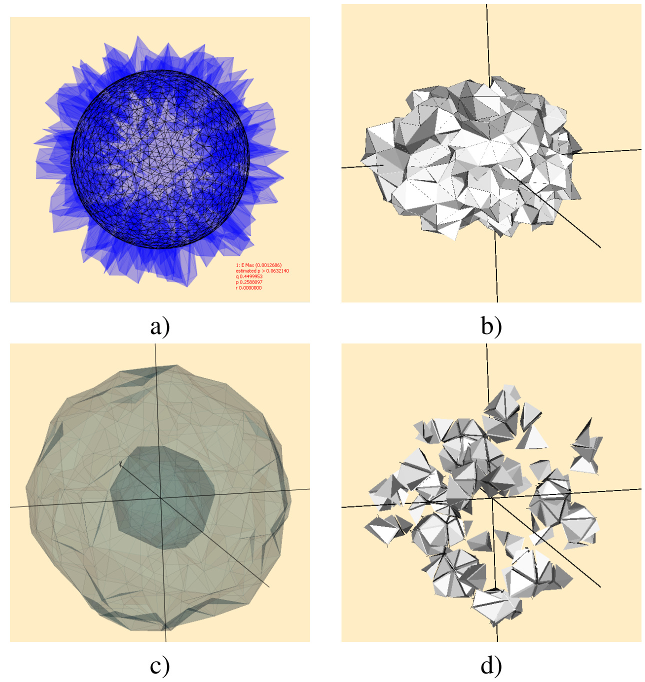


Figure 1. Surface of a unit sphere. The data set consists of 1001 nearly evenly distributed sample points with $q = 0.45$

4 Implementation and Timings

In the table below we show the breakup of the timings of the thinning reconstruction algorithm for a number of data sets on PC with 2000Mhz "Intel Centrino Duo" CPU and 2GB memory. The data sets [2, 1] were downsampled in order to show that our method does not require high resolution sampling for topologically correct surface reconstruction. The experiments were performed twice i.e. first with noise-free data, second with noise corrupted data. The corresponding p, q-values are to be found in the table. The corresponding results are demonstrated in Figures 5. As can be seen, the topology can even be reconstructed in cases when the geometry is heavily distorted by the noise. Please notice, that our implementation of the algorithm was designed for demonstration purposes. Thus, the timings are not to be considered to be optimal.

data set	points	p	q	α-shape millisec	thinning millisec
Skeleton Hand	29176	0.045 0.047	0 0.01	33828 36921	18859 18452
Dragon	28395	0.0026 0.0026	0 0.0011	32265 25578	13390 20297
Armadillo	34006	1.81 2.24	0 3.61	32226 57605	15749 37101
Angel	31478	0.22 0.27	0 0.18	36591 28158	15719 33736
Knots	23232	0.013 0.015	0 0.012	25141 25323	10969 18637

We divided the reconstruction procedure into two steps The "α-Shape" builds the data structure by computing the Delaunay Triangulation and fills the data structure, "α-Shape" computation, which stores for every element of the data structure the α-value and the Thinning procedure, which performes the topologically correct surface reconstruction.

"CGAL" [3] library and its build in data structure was used for sampling represention, computation of the Delaunay triangulation and the "α-shape". As it is descriped in the algorithm we have to delete simple cells from the reconstruction. For complexity reasons we used labels to classify the appropriate elements of the Delaunay Triangulation instead of changing the entire data structure. Thus, in the process of "deletion" we just remove the labels.

We do not recommend to use the build-in function of sorting the triangulation elements in a list and use this list as a data structure. Though the usage of "filter iterators" is tempting for clean coding, the implementation of the required predicates is inefficient. We observed, that the thinning procedure required fourfould of the time given in the table above.

Since the worst case complexity of computing α-shapes in 3D is $O(n^2)$, and efficient priority queue implementations need $O(\log n)$ time both for push and pop, we get a worst case complexity of $O(n^2 \log n)$ for our algorithm. Nevertheless in case of typical surface sampling data we observe subquadratic time complexity.

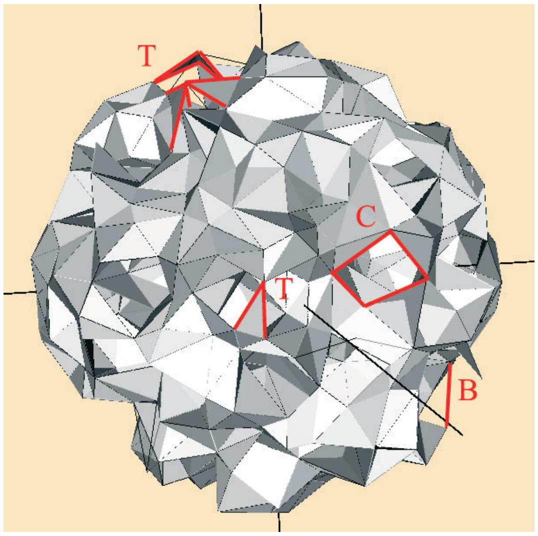


Figure 2. Marked are some of irregularities of the α-shape: concavities (C), tunnels (T), bridges (B)

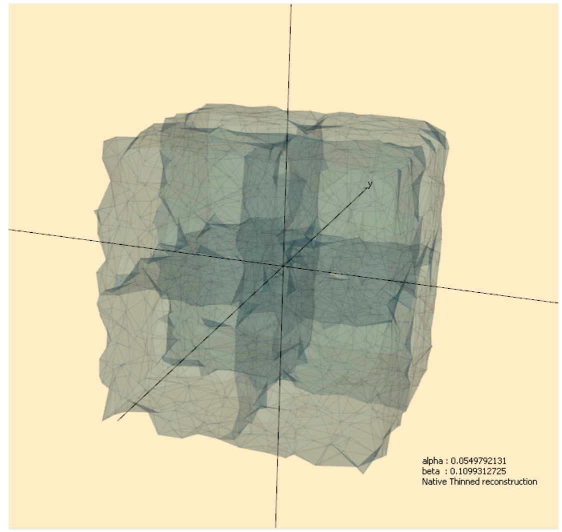


Figure 3. Reconstruction of a cube containing 8 regions

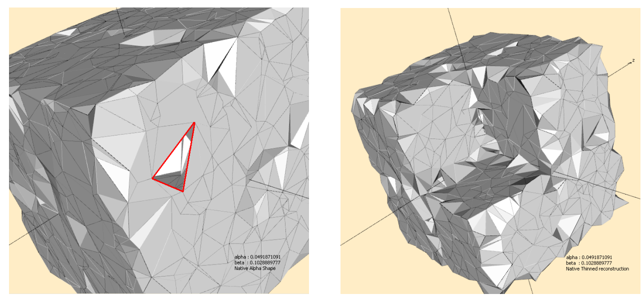


Figure 4. Surface Reconstruction in case of undersampled boundary

5 Experimental Evaluation

First we are going to investigate which problems our algorithm has to solve due to the high amount of noise and also due to the fact that we allow non-smooth surfaces. To illustrate the subject matter we create a sampling data set corrupted by a great amount of noise. We sampled the surface of the unit sphere and added $q = 0.45$ error to the sample points. The original noise-free sampling is demonstrated as black triangulation in figure 1a, while the noisy triangulation is shown in blue. The result of surface reconstruction is shown in b. Geometrically it lies inside the dilation of radius q of the original surface (which is approximated in c), although the geometrical distortion is relatively high due to the linear dependence on the amount of noise. As shown in d, the α-shape contains cavities (spurious holes), which are filled by our algorithm. Figure 2 demostrates the $alpha$-shape of the surface. Some of the irregularities of the are marked i.e. α-shape concavities (C), tunnels (T), bridges (B).

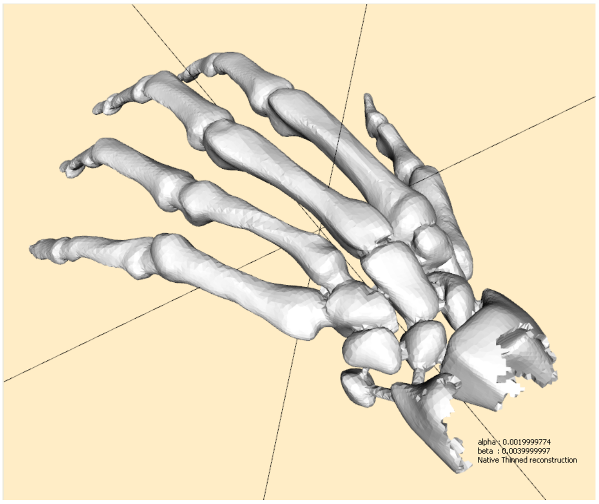
Figure 3 demonstrates the ability of our method to reconstruct the boundary of non-smooth and non-binary partitions with junctions and corners. We divided a cube into 8 equal regions by three intersecting squares. As it may be observed in Figure 3 the regions could be correctly reconstructed.

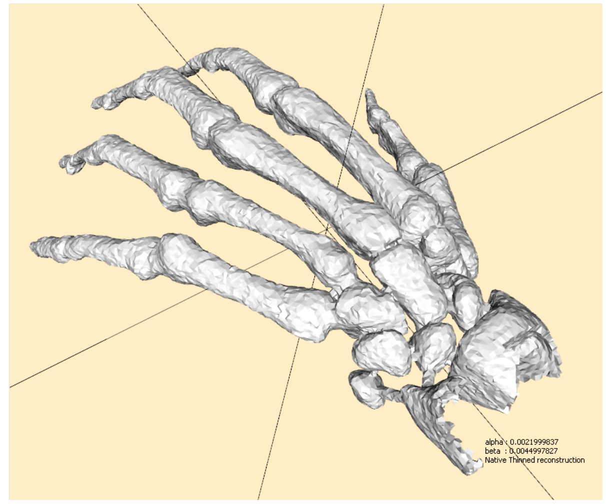
6 Conclusions and Future Work

Our method results in topologically equivalent components of the boundary. The given bounds are much better than previous ones, although in reality one often has to deal with even worse samplings. In fact in most of our experiments we had to estimate the values of p and q empirically, but the algorithm behaved well even in cases where especially the bound given for γ was not fulfilled. Moreover for the first time it is possible to guaranty a topologically correct boundary reconstruction in case of non-smooth surfaces and in case of partitions into more than two regions. Nevertheless our approach is not robust to undersamplings, since the thinning algorithm strictly requires a sufficiently dense sampling of the whole surface. The result of such reconstruction may be seen in figure 4. The top left picture shows the undersampled part of the region which results in a hole in the surface. The thinned reconstruction is demonstrated in the top right picture. We see that the reconstruction containes only 7 regions du to the fact that the boundary of one region is completely removed by the thinning algorithm. In future we will try to overcome this problem. In addition to that we will try to adapt our method to adaptive sampling schemes.

Acknowledgment

The authors thank the Deutsche Forschungsgemeinschaft (DFG) for funding via project STI 147/2-1.





References

[1] 3D Model Retrieval. http://3d.csie.ntu.edu.tw/~dynamic/cgi-bin/DatabaseII_v1.8/index.html.

[2] Large Geometric Models Archive. http://www-static.cc.gatech.edu/projects/large_models/.

[3] CGAL, Computational Geometry Algorithms Library. http://www.cgal.org.

[4] N. Amenta, M. Bern, and M. Kamvysselis. A new voronoi-based surface reconstruction algorithm. *Conf. on Comp. Graph. and Interact. Techn.*, pages 415–421, 1998.

[5] N. Amenta, S. Choi, T. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. *Symp. on Comp. Geom.*, pages 213–222, 2000.

[6] N. Amenta, S. Choi, and R. Kolluri. The power crust, unions of balls, and the medial axis transform. *Comp. Geom.*, 19(2):127–153, 2001.

[7] J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of lipschitz surfaces. *Proc. 22th Annual ACM Symposium on Computational Geometry*, 2006.

[8] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Proc. 21th Annual ACM Symposium on Computational Geometry*, pages 263–271, 2005.

[9] T. Dey, H. Edelsbrunner, and S. Guha. Computational topology. *Advances in Discrete and Computational Geometry*, 223:109–143, 1999.

[10] T. Dey and S. Goswami. Tight cocone: A water-tight surface reconstructor. *J. Comp. and Inf. Sci. in Engin.*, 3:302, 2003.

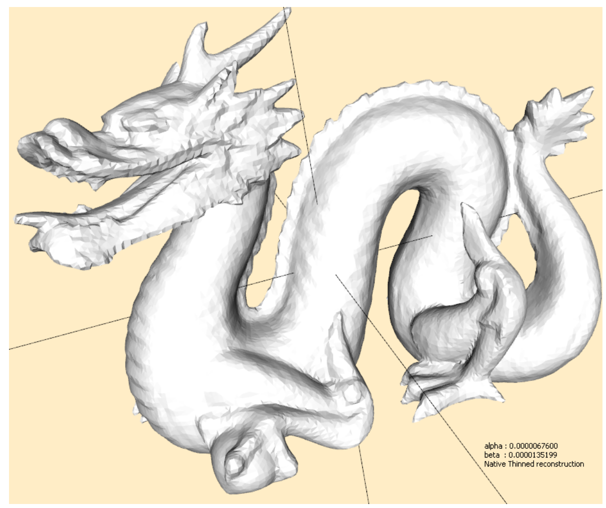
[11] B. Mederos, N. Amenta, L. Velho, and L. de Figueiredo. Surface reconstruction from noisy point clouds. *Eurographics Symp. on Geom. Proc.*, 2005.

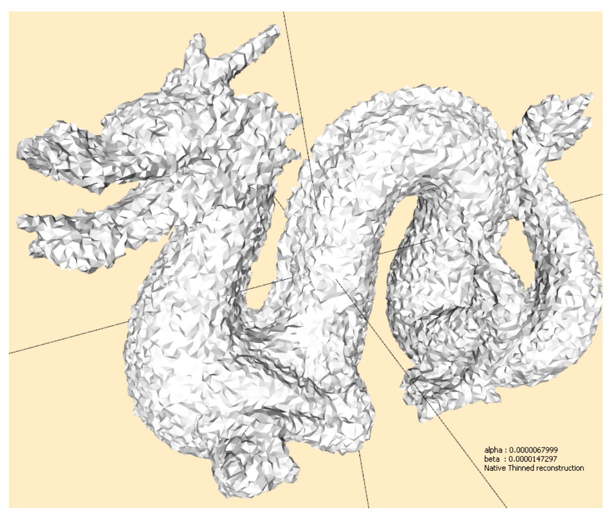
[12] P. Niyogi, S. Smale, and S. Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discr. and Comp. Geom.*, 39(1), 2008.

[13] P. Stelldinger. Tight bound for sphere covering and extension to r-regular sets. *Submitted*.

[14] P. Stelldinger. Topologically correct 3d surface reconstruction and segmentation from noisy samples. *Int. Worksh. Comb. Img. Anal., IWCIA*, 2008.

[15] P. Stelldinger. Topologically correct surface reconstruction using alpha shapes and relations to ball-pivoting. *Submitted*, 2008.





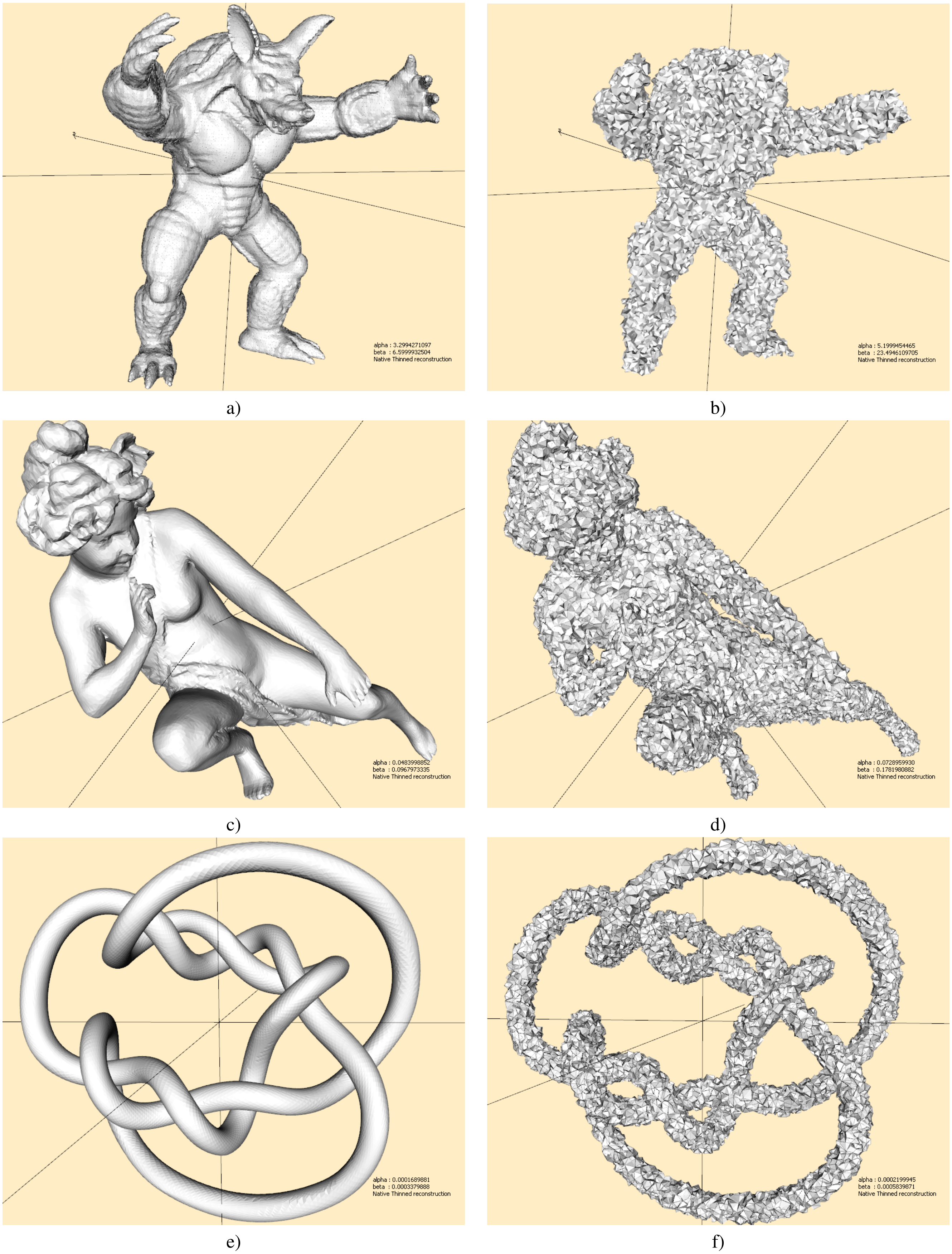


Figure 5. Noise-free and noise corrupted reconstruction