FitConnect: Connecting Noisy 2D Samples by Fitted Neighborhoods

S. Ohrhallinger[1] and M. Wimmer[1]
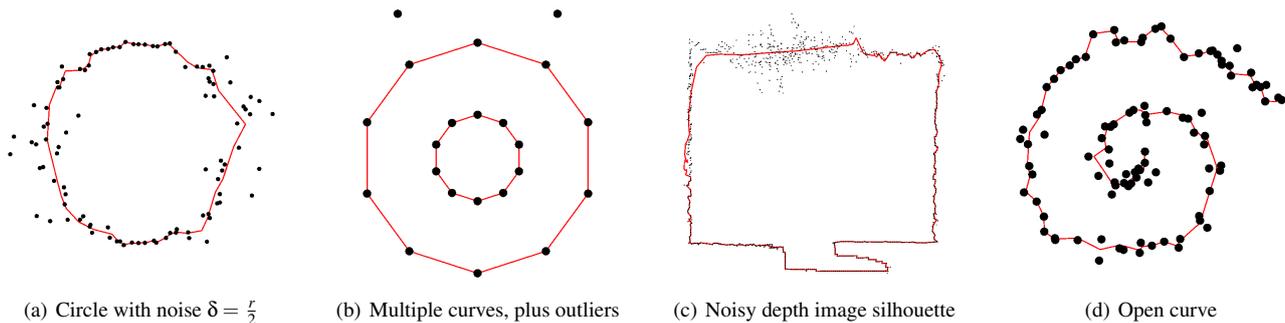

(a) Circle with noise $\delta = \frac{r}{2}$ (b) Multiple curves, plus outliers   (c) Noisy depth image silhouette   (d) Open curve

Figure 1: *Our parameter-free method reconstructs open and closed curves from 2D samples with highly varying noise and outliers.*

Abstract

We propose a parameter-free method to recover manifold connectivity in unstructured 2D point clouds with high noise in terms of the local feature size. This enables us to capture the features which emerge out of the noise. To achieve this, we extend the reconstruction algorithm HNN-CRUST*, which connects samples to two (noise-free) neighbors and has been proven to output a manifold for a relaxed sampling condition. Applying this condition to noisy samples by projecting their* k*-nearest neighborhoods onto local circular fits leads to multiple candidate neighbor pairs and thus makes connecting them consistently an NP-hard problem. To solve this efficiently, we design an algorithm that searches that solution space iteratively on different scales of* k*. It achieves linear time complexity in terms of point count plus quadratic time in the size of noise clusters. Our algorithm* FITCONNECT *extends* HNN-CRUST *seamlessly to connect both samples with and without noise, performs as local as the recovered features and can output multiple open or closed piece-wise curves. Incidentally, our method simplifies the output geometry by eliminating all but a representative point from noisy clusters. Since local neighborhood fits overlap consistently, the resulting connectivity represents an ordering of the samples along a manifold. This permits us to simply blend the local fits for denoising with the locally estimated noise extent. Aside from applications like reconstructing silhouettes of noisy sensed data, this lays important groundwork to improve surface reconstruction in 3D. Our open-source algorithm is available online.*

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Reconstructing curves from noisy unstructured points in 2D is a fundamental problem that has been studied extensively in computer graphics and computational geometry and is an important base for surface reconstruction. It has applications in reverse engineering of geometric models, e.g., reconstructing object boundaries from sensed data, such as 2D slices of 3D data, or segmentation from silhouettes of depth images, and as such is even important in 2D. However, the general approach of trying to recover arbitrary geometry when not using priors is ill-posed. Boundaries of physical objects are manifold by nature, but may be partially oc-

cluded, contrary to their projections a plane, which may then intersect or contain T-junctions. Clustering points of similar depth from depth images prevents this merging into non-manifold connectivity. Thus, we restrict the problem slightly, to recovering manifold curves (open or closed), so that we can solve it by robustly computing the local inside/outside of the curve. Further, we aspire to recover all features which are not hidden by the extent of the noise.

Prior work manages to produce (smoothed) curves from noisy point sets, but is often restricted, e.g. to outputting a single open curve, to globally uniform noise, or does not recover the connectivity in a reproducible fashion. What is missing yet is a method which

relates the reconstruction distance to the distance of noisy samples from the ground truth and also has reasonable time complexity to produce output within acceptable run-time.

As noisy samples are always based on the original undistorted signal, it is imperative to relate them to the assumed boundary we want to recover. It is also clear that only features of larger extent than the local noise provide information to become extracted.

For a smooth curve, the extent of the noise can be expressed in terms of the local feature size (*lfs*). We put forward the axiom that those features for which the noise extent is small w.r.t. its *lfs* can be recovered. Based on this axiom, we claim that the noise is small everywhere, compared with the relevant (because recoverable) features. Clusters of noisy points can be approximated by local manifolds, e.g. circular arcs, as long as their curvature inside their neighborhood is small. However, since the extent of the local noise is not known, the challenge remains to determine the correct neighborhood size for these fits, which cannot be done locally. Since we assumed previously that the curve is a (bounded) manifold, the locally fitted curves have to be consistent. By adjusting the neighborhood size for the locally fitted curves such that they match among each other, we can recover the manifold connectivity, its orientation and an ordering of the samples. The size of the locally fitted neighborhoods and the location of their samples give an estimate of the local noise, therefore we can use it for denoising as a post-process.

Our major contribution is the algorithm FITCONNECT, which robustly recovers manifold connectivity from arbitrarily noisy 2D samples, that is features emerging over the noise extent. Applying multi-scale to local searching is an approach we have not seen yet in related work. It enables both reducing the combinatorial complexity of the search space and parallel local computations.

The major features of FITCONNECT are:

- Seamless extension of HNN-CRUST to handle noisy samples.
- Detect noisy samples and the extent of local noise.
- Simplified output by representative points of noisy clusters.
- Consistently ordered local neighborhood fits create a manifold.
- Simple blending of local fits to approximate the original curve.
- Time complexity log-linear in the number of points and squared in the size of noisy clusters.
- Robust – we provide open source to support this claim.

Our curve reconstruction lays important groundwork for surface reconstruction since the geometric concepts used (sampling condition, circular fitting, local manifoldness) extend well into 3D.

2. Related Work

We first take a look previous work, for reconstruction of curves from noise-free samples, noisy data sets and curve simplification.

Reconstruction from Noise-free Samples Ohrhallinger et al. [OMW16] give a detailed overview of the evolution of these reconstruction algorithms which are often based on sampling assumptions. Starting by requiring uniform sampling density [EKS83, KR85, FMG94, Att97, DT14, DT15, Ste08, ST09], Amenta et al. [ABE98] introduced the ε-sampling condition based

on the local feature size (lfs) which spurred further development [Gol99, DK99, Alt01, Len06, PM16], extending it to handle open curves [DMR99], sharp corners [DW02, FR01] and modifying the sampling condition to get tighter bounds [OM13, OMW16].

Applications and Reconstruction from Noisy Samples Birkas et al. [BBP16] show a system for retrieving objects from mobile sensed data by segmenting them via clustering. From these point clusters, (partially occluded) silhouettes can be extracted, which are noisy due to sensor artifacts.

Compared to the well-structured problem of reconstruction from noise-free samples, there is considerably less prior work for unstructured noisy samples and none that directly applies to our stated problem. DeGoes et al. [DGCSAD11] attempts to solve a larger problem which also includes intersecting curves. They construct the Delaunay triangulation of the point set, then greedily simplify it to minimize transport cost. For the problem domain we consider, it performs well for uniform noise, dense outliers and close curves but fails to connect samples of variable density as a global parameter controls the amount of simplification. Consequently, our experiments below show that its behavior is very sensitive to this specified iteration count and that it has difficulties to reconstruct manifold outputs for non-uniform sampling or high local noise. A similar method [WYZ*14] fails to reconstruct curves from moderately sparse point sets as well, as can be seen in their Fig. 28. Our algorithm handles such sparse point sets without noise well since it behaves like HNN-CRUST, which it extends seamlessly, as can be seen in the results [OMW16].

As mentioned in the introduction, we aim to solve the simpler problem of extracting just manifold curves, which is also the most interesting aspect. Mehra et al. [MTSM10] apply a visibility operator using the convex hull to extract local connectivity which then is combined globally in a weighted graph. They propose a fast approximation algorithm to extract the maximum weight cycle. It does not denoise and produces gaps even for low extents of noise. The method of Lee [Lee00] is restricted to reconstructing a single open curve. This is due to using the Euclidean Minimum Spanning Tree to recover the connectivity. They further smoothen the reconstructed manifold with Moving Least Squares and fit a spline curve. Contrary to above work, our method can output open and closed, as well as multiple connected curves and most importantly, handles variable sampling density and/or noise extent well. Poisson reconstruction [KH13] requires reliable normals for reconstructing noisy point clouds [ACSTD07] while we consider unstructured samples.

Dey and Goswami [DG06] prove an existence result, for a proposed noise model that restricts the noise to an unspecified fraction of the local feature size. Another existence results is proven by Cheng et al. [CFG*05], for reconstructing a curve with probability as a function of sample noise w.r.t. the local feature size, but impractical $O(N^3)$ time complexity. Their algorithm grows a disk neighborhood around a sample until it fits a strip with relatively small width. Then, samples are decimated and connected using the noise-free reconstruction algorithm NN-CRUST [DK99]. Our method also assumes a noise model that expresses noise as a function of the local feature size but delivers practical results within reasonable runtime. On top of that, the recovered connectivity represented by locally overlapping fits captures the properties of the

local noise. We simply blend these local fits, with already good results, but more effective denoising methods are available, e.g. Feiszli and Jones [FJ11] show how a manifold curve can be denoised effectively by multiscale analysis and a corner detector.

The rest of the paper is organized as follows: In Section 3 we state some definitions in order to revisit the algorithm HNN-CRUST and then derive the conditions for connecting samples contaminated by noise. Our algorithm for searching the solution space efficiently is described in Section 4. We present results of the curve reconstruction, evaluation w.r.t. to ground truth and further analysis in Section 5. Section 6 concludes with an outlook to future work.

3. Definitions

We reproduce the following definitions [OMW16]:

C is a collection of *smooth curves*, which are (possibly bounded) 1-manifolds embedded in $\mathbb{R}^2$ and twice-differentiable everywhere except at boundaries. C can thus consist of one or more connected simple curves, i.e., loops and segments (bounded by two *terminus* points), prohibiting T-junctions or crossings. Since the curves are smooth, they cannot contain sharp angles.

Let $\|d\|$ denote the Euclidean L_2 norm. Each sample $s \in S$ is within Euclidean distance of a point $p \in C$ such that $\|p,s\| < \varepsilon(p)$. We define the *nearest neighbor* s_0 to a sample point s_1 as $\arg\min_{s_j \in S \setminus s_1} \|s_1, s_j\|$. Further, the *half neighbor* s_2 is the closest sample in the half-space H which is partitioned by the perpendicular bisector of the edge $\overline{s_0 s_1}$ and does not contain s_0: $\arg\min_{s_j \in S \setminus s_1, s_j \in H} \|s_1, s_j\|$. Let n_i be the i-th nearest sample to s_1 by Euclidean distance. Let $N(s)$ be the set of neighborhood points of s, for an unspecified k such that it is equal to $N_k(s)$, the set of k-nearest neighbors of s, $N_0(s) = s$.

We first revisit the algorithm HNN-CRUST [OMW16], which reconstructs curves by interpolating samples (assumed to be noise-free). Then we describe its properties and show how we can extend it seamlessly to samples contaminated with noise.

3.1. HNN-CRUST revisited

HNN-CRUST connects each sample $s \in S$ to its nearest neighbor n_0 and its half neighbor n_h (in case s is not a terminus of the curve). Let h be the perpendicular bisector of the nearest neighbor edge, then n_h lies in the half-space containing s (see Figure 2). This leads to (see Figure 3a):

Condition 1 (Conformity): The distance between the two neighbors $d = \|n_0, n_h\|$ is larger than their respective distance to s: $d > \|n_0, s\| \wedge d > \|n_h, s\|$.

Condition 1 permitted to prove that HNN-CRUST reconstructs the curve under a very relaxed as well as close-to-tight sampling condition ($\rho < 0.9$, equivalent to $\varepsilon < 0.47$) [OMW16]. For this sampling condition, the reconstructed curve has to be a (bounded) 1-manifold that connects the samples $s_i \in S$ in their order on C. That means that for a sample s_i which is not a terminus, the point pair (s_{i-1}, s_{i+1}) corresponds to (n_0, n_h), ordering notwithstanding. Let $T(s)$ be the tupel of two neighbors (or one, in case of a terminus) for a sample s. Then it follows (see also Figure 3b):
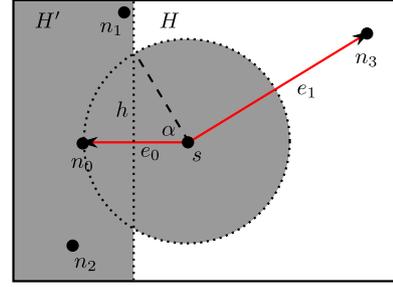

Figure 2: HNN-CRUST *reconstruction of an edge-pair for a sample s (image courtesy of [OMW16]). Edge e_0 connects s to its nearest neighbor n_0. The other edge e_1 is the shortest edge connecting s with a vertex in halfspace H. Further, observe that this vertex (here n_3) must lie inside the white shaded area of H, since no sample is closer to s than n_0. This implies an angle $\geq 60°$ between both edges and that both neighbors are further apart from each other than from s.*
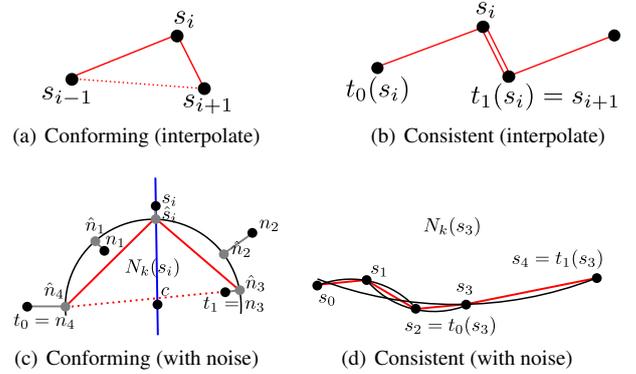

Figure 3: *Top row: The two conditions for ensuring consistency between conforming neighbors for interpolating samples (HNN-CRUST). c): Projections of the k-neighborhood $N_k(s_i)$ onto its circular fit yields the representative neighbors for Condition 3 as those farthest per half-space spanned by the line $(s_i, \overrightarrow{cs_i})$. d) Condition 4 requires a consistent ordering of these representative neighbors within each $N_k(s_i)$ for all $N_k(s_j)$ overlapping with it.*

Condition 2 (Consistency): $s_j \in T(s_i)$ is a commutative operator, and $s_i \in T(s_j)$.

We will proceed to show how we can exploit these two conditions in order to connect samples in the presence of noise, and extend them to formulate our proposed algorithm FITCONNECT.

3.2. Extension of HNN-CRUST conditions to noisy samples

We assume that the noise extent of the samples is unknown and can vary locally. Therefore, we have to determine a local approximation of the curve at a sample s. If the neighborhood $N_2(s)$ does not fulfill Condition 1, we *classify s as noisy w.r.t. its neighborhood* and have to extend $N_s(s)$ to $N_k(s)$ until the local approximation represents a

feature emerging over the noise extent. Note that the three points in $N_2(s)$ define a circle. For an increasing neighborhood $N_k(s)$, we approximate the curve locally by fitting a circle such that the distances to the samples are minimized in the least-squares sense. This circular fitting of curves (similar to a spherical fitting of surfaces) works well, as has already been demonstrated here [GG07], by robustly computing the local inside/outside of the curve.

For circular fitting, either geometric or algebraic methods can be used. Geometric fitting results in a non-linear least squares problem, which can be approximated with iterative methods like Gauss-Newton. While they are very accurate and correspond to the maximum likelihood estimation for the circle parameters, they are also computationally expensive and may have bad convergence properties. For our circular fit we therefore choose an algebraic method, because it can be formulated as a constrained optimization problem and then solved as an Eigenvalue problem [CL05]. We select the HYPER-FIT technique, which has *zero essential bias* and has been shown to be even more precise then geometric methods [ASC*09].

It yields the fitted circle $\Gamma_{s,k}(c,r)$ for $N_k(s)$, onto which we can project all neighbor samples $n_i \in N_k(s)$ as $\hat{n}_i = c + r\overrightarrow{cn_i}/\|\overrightarrow{cn_i}\|$, with c being the center of $\Gamma_{s,k}$ and r its radius. For each half-space spanned by the line $(c,\hat{s}), \hat{s} = \hat{n}_0$, we select the representative neighbor $t_{[0,1]} = \hat{n}_i$ such that $\hat{n}_i$ is farthest from $\hat{s}$ (see Figure 3c). Now, using $T(s) = [t_0, t_1]$ we can reformulate Condition 1 for the local fit of a noisy neighborhood of s as follows:

Condition 3 (Conformity): The distance $d = \|t_0, t_1\|$ between the two neighbors of s is larger than their respective distance to s: $d > \|t_0, s\| \wedge d > \|t_1, s\|$.

Applying Condition 2 directly to $T(s)$ is not sufficient since the neighborhoods now can contain other samples than the single neighbor per side. A consistent ordering of these samples has to be ensured to guarantee a manifold. We define an ordered set $O(S, N)$ that orders the subset $S \subseteq N_k(s_i)$ over its containing neighborhood $N_k(s_i)$ conforming to Condition 3, such that it collects the points $s_k \in S$ projected to $\Gamma_{s_i,k}$ as $\hat{s}_k$ starting from t_0, moving in the direction of s_i to finish at t_1. Then we can formulate this consistent ordering as (see Figure 3d):

Condition 4 (Consistency): $\forall s_j \in N(s_i), S = N(s_i) \cap N(s_j) : O(S, N(s_i)) = O(S, N(s_j))$, which is also commutative.

Note that incrementally growing orderings of commutative subsets do not necessarily have to contain each other, as in $O(S, N_k(s_i)) \subset O(S, N_{k+1}(s_i))$. We only require different adjacent neighborhoods to be consistent, but not at a specific scale, since we aim to find some valid ordering rather than to solve the NP-hard problem. Our experiments show the determined solutions to be close to optimal.

Since three samples define a circle (or two samples a line as degenerate circle in case of a terminus), the Conditions 3 and 4 form a superset of Conditions 1 and 2 respectively and thus extend HNN-CRUST seamlessly to the locally fitted circles of noisy samples.

Condition 4 gives an important guarantee for the reconstructed manifold w.r.t. the samples as it creates an ordering of these through the locally consistent neighborhoods. This enables anistropic denoising, as shown later on, and might lead to more applications.

4. Algorithm FITCONNECT

The connectivity reconstruction for interpolating samples is straightforward and of linear time complexity, since both neighbors for each sample and the consistency between neighbors can be determined very simply. However, as stated in the introduction, varying noise cannot be detected just locally. Small neighborhoods below the actual noise extent could be conforming to Condition 3, but not to Condition 4, since inconsistent with their neighbors' local fits. Each sample s potentially has a number of conforming local fits for $N_k(s)$ of varying k. For each of these, the consistency among all samples $s_i \in S$ would have to be tested, in order to determine a consistent manifold for those locally conforming fits. Since up to N local fits per sample would have to be tested among N samples, an exhaustive search of the solution space becomes an NP-hard problem. In order to search the solution space efficiently, we therefore propose an iterative approach that turns out to work well. In short, FITCONNECT differs from HNN-CRUST insofar as it first locally estimates the feature size and then applies the connectivity conditions of HNN-CRUST.

4.1. Efficiently Recovering Closed Manifold Connectivity

For easier understanding, we first start with the simplified assumption that the samples represent a manifold without boundaries. Basically, Algorithm 1 starts for each sample with its $k = 2$ nearest neighbors and iteratively increases k for each sample which is not conforming (Condition 3) or consistent (Condition 4) with its neighbors. This should let our algorithm converge quickly to a valid – if not optimal – solution.

ALGEBRAICCIRCULARFIT constructs local manifolds quickly for samples with no or little noise. We use the conforming fit (testing Condition 3 in ISCONFORMINGFIT) of the samples with the most k neighbors recorded while iterating to connect with consistent neighbors (Condition 4). If there are several choices to connect a neighbor s_j to s_i, we select the one farthest on $\hat{N}(s_i)$ to s_i in UPDATEFARTHESTCONSISTENTNEIGHBORS, which also mutually updates s_i as neighbor of s_j. Before assigning these new neighbors, we have to remove the current ones, also reciprocally, which is done in REMOVENEIGHBORSMUTUALLY. This maximizes the overlap and reduces complexity without losing topology, since the fitted neighborhood represents only a single feature. See Fig. 4 for an illustrative example of adding and removing edges to the manifold.

However, the neighborhood of a sample which already overlaps on one side with its neighbor may become much larger while trying to locate a consistent neighbor on its other side, due to locally varying sampling density or noise extent. In our experimentations this led to entirely covering and eliminating smaller neighborhoods, together with the features they represented, resulting in a valid but suboptimal solution. To avoid that, we keep existing neighbor relations, but require that all successively included neighbors of s in each half-space of $N(s)$ are also ordered in the same fashion on $\hat{N}(s)$, as tested in CONSISTENTNEIGHBORHOOD (Condition 4).

INTERLEAVEDNEIGHBORS handles the rare case that neighbor points cannot be consistently connected because projections on their mutual neighborhood fits interleave, by simply removing
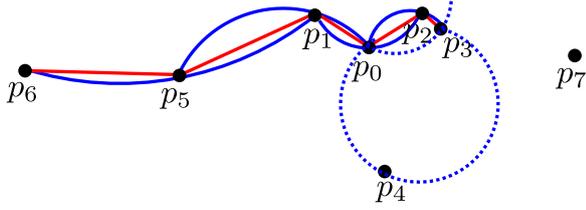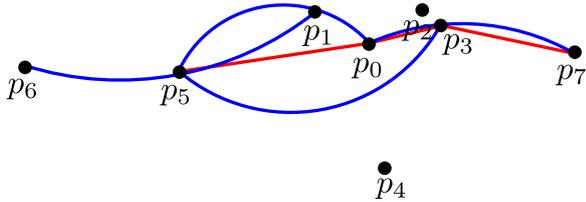
(a) Iteration #1: p_5, p_1, p_0, p_2 conforming and consistent within N_2, while for p_3, p_4 their N_2 is not conforming to Condition 2.
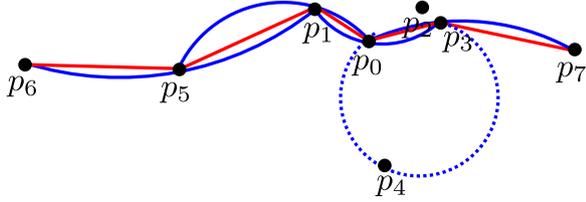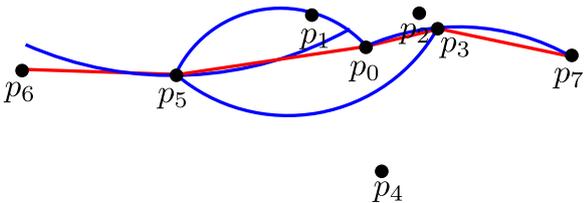

(b) Iteration #2: p_3 becomes conforming, eliminates redundant p_2
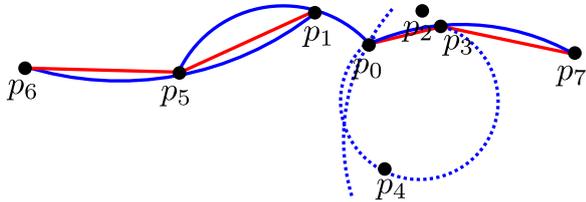
(c) Iteration #3: $N_4(p_0)$ becomes (and p_4 remains) not conforming

(d) Iteration #4: $N_5(p_0)$ becomes conforming and p_5 inconsistent

(e) Iteration #5: $N_3(p_5)$ becomes conforming and consistent again

Figure 4: *Illustrative example of iteratively converging to consistent edges between conforming vertices: a) Each point considers its two nearest neighbors N_2. b) The not conforming p_3, p_4 invalidate p_0, and all three increase their neighborhood to N_3 (three points). c) The increased $N_4(p_0) = \{p_1, p_2, p_3, p_4\}$ becomes not conforming. d) Adding p_5 to $N_5(p_0)$ makes it conforming again but inconsistent with p_5 since its end vertex p_1 is eliminated. e) Increasing to $N_3(p_5)$ makes p_5, p_0 consistent again, resulting in a manifold.*

one of these (close) points. Furthermore, points which are shared between neighbors (as determined between new neighbors, or in EDGEEXISTSIN) are considered redundant and eliminated, as they project onto already constructed edges in the manifold and do not contribute to features (Figure 5).

We expect the time complexity to be $O(N \log N)$ due to the kd-tree construction and squared in the sizes of fitted neighborhoods.

Algorithm 1 Reconstructing manifold connectivity in noisy points

---

Input: S ▷ The set of samples
Output: P ▷ Subset of connected output points $P \subseteq S$
Output: $T : P \mapsto P \times P$ ▷ Neighbor tupels of P

1: $P \leftarrow S$
2: $T \leftarrow \{\}$
3: $N(p) \leftarrow N_1(p)$
4: **while** $\exists p \in P : |T(p)| < 2$ **do**
5:       $N(p) \leftarrow N_{|N(p)|}(p)$ ▷ Add next nearest neighbor
6:       $\Gamma_p \leftarrow$ ALGEBRAICCIRCULARFIT$(N(p))$
7:       $Q \leftarrow \{\}$ ▷ Track points affected by updates
8:       **if** ISCONFORMINGFIT(Γ_p) **then**
9:           $Q \leftarrow Q \cup T(p)$
10:          REMOVENEIGHBORSMUTUALLY(T, p)
11:          UPDATEFARTHESTCONSISTENTNEIGHBORS(T, p)
12:          $Q \leftarrow Q \cup T(p)$
13:          **for** $q_i \in N(p) \setminus T(p)$ & $q_i \in N(T(p)) \setminus p$ **do**
14:              $P \leftarrow P \setminus q_i$ ▷ Eliminate redundant points
15:              $Q \leftarrow Q \cup q_i$
16:          **end for**
17:          **if** INTERLEAVEDNEIGHBORS$(N(p), T(p))$ **then**
18:              $P \leftarrow P \setminus p$ ▷ Eliminate obstacle point
19:              $Q \leftarrow Q \cup p$
20:          **end if**
21:      **else**
22:          **if** EXISTSEDGEIN$(N(p), T)$ **then**
23:              REMOVENEIGHBORSMUTUALLY(T, p)
24:              $P \leftarrow P \setminus p$ ▷ Eliminate redundant points
25:              $Q \leftarrow Q \cup p$
26:          **end if**
27:      **end if**
28:      **while** $Q \neq \{\}$ **do** ▷ For all points affected by updates
29:          Select a q from Q
30:          $Q \leftarrow Q \setminus q$
31:          $Consistent \leftarrow True$
32:          **for** $n_i \in P \wedge q \in N(n_i)$ **do** ▷ Test all referencing points
33:              **if** CONSISTENTNEIGHBORHOOD$(n_i) =$ False **then**
34:                  $Consistent \leftarrow False$
35:              **end if**
36:          **end for**
37:          **if** $Consistent =$ False **then**
38:              REMOVENEIGHBORSMUTUALLY(T, q)
39:              $Q \leftarrow Q \cup q$
40:          **end if**
41:      **end while**
42: **end while**

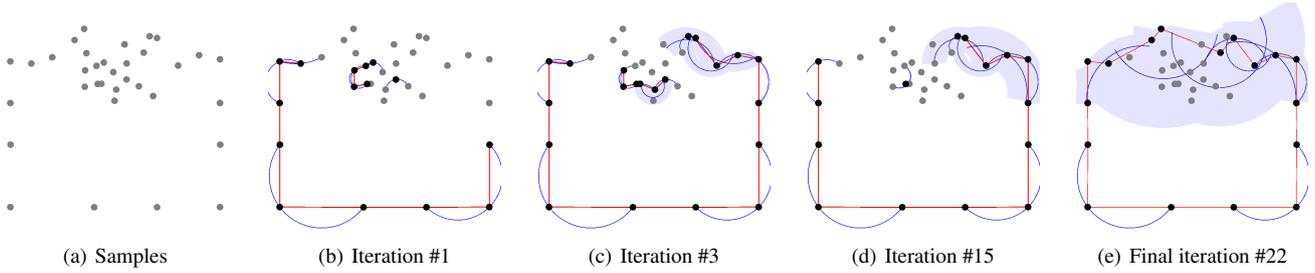(a) Samples   (b) Iteration #1   (c) Iteration #3   (d) Iteration #15   (e) Final iteration #22

Figure 5: RECTANGLE*: a) Samples of a rectangle (top boundary highly corrupted by noise). b) The first iteration connects samples' nearest two neighbors if they conform (blue circular fits for black points). c) Connectivity is extended by fitting noisy samples (noise extent of local fits shaded light blue). d) Inconsistent local connectivity gets removed and partially replaced with larger fits. e) Final reconstructed manifold connectivity (red edges connecting black points) leaves just a few points in the noisy cluster on the top, which have large neighborhoods.*

4.2. Handling Manifolds with Boundaries

In order to also handle manifolds with boundaries, we have to detect them using a suitable fitting criterion. There is no inherent distinction whether samples are just sparsely spaced or represent a hole, so we have to introduce an artificial differentiating criterion which gives acceptable results. We propose that a hole should not be larger than a nearby feature. Thus we do not relate our criterion of hole size just to relative sample density but to the captured feature size, represented by the size of the local neighborhood fit. We classify a sample s as a terminus of the curve if it only contains points in one half-space of its fitted neighborhood. It has to have two $N_k(s)$ neighborhoods as follows: one with two successive consistent points to avoid trivial solutions, and a larger size neighborhood that is inconsistent with that feature (or $k = N$ in the limit for small examples without features). Examples for reconstructed open curves are shown in Figures 8c and 17.

4.3. Handling Sharp Corners

Our sampling condition requires projected open angles of $> 60°$ (Condition 3), which places a severe limitation on the input class. Sharp corners would therefore be rounded off until our operator detects a fit when the opening angle becomes large enough. In the worst case, it would not reconstruct contrived cases such as a rhombus with too small angles (see Figure 18a). To enable successful reconstruction in these cases, we detect sharp corners by their incident manifold neighborhood. Our sharp-corner detector considers the neighborhood of a point whose fit does not conform. If its neighborhood disc contains exactly two curves which are open and intersect the boundary of the disc at an angle of $\leq 60°$, we connect these two open curves to the current point, provided that this creates no self-intersections. Applying this sharp-corner detector successfully reconstructs the point configurations in Figures 7g and 18a.

4.4. Simple Blending to Approximate the Original Curve

The output of Algorithm 1 extended by the boundary handling is a (possibly bounded, consisting of multiple connected components) manifold $M(P, T)$ which connects a subset of output points $P \subseteq S$ by edges to their neighbor relations T. However, in case the samples are contaminated by noise, this interpolating piece-wise curve
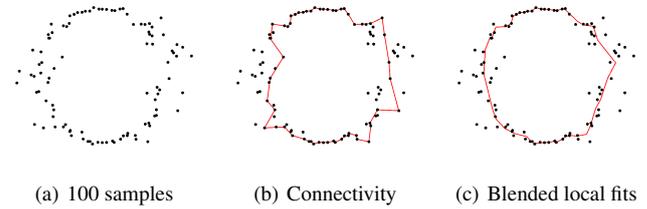

(a) 100 samples   (b) Connectivity   (c) Blended local fits

Figure 6: CIRCLE*: a) 100 samples with noise δ varying between 0 (top, bottom) and 0.5r (sides) of the circle. b) The reconstructed connectivity oscillates for high noise extent. c) Blending the local fits results in a much better approximation of the original curve.*

may not be a good reconstruction of the original curve in terms of geometric closeness. The neighborhoods of the local fits with size $|N(p)| > 2, p \in P$ indicate the presence of noise and the variance of $N(p)$ represents its extent. Therefore, we perform a simple postprocessing step by blending the circular fits of the neighborhoods including p for each p classified as noisy.

To compute the blended point p_i' for a $p_i \in P$, we consider all fitted circles Γ_j of $p_j \in P$, where $p_i \in N(p_j)$, and whose arcs $a_j(\hat{t}_0, \hat{t}_1) \subset \Gamma_j$ intersect the normal of the local fit of p_i. Due to our Condition 4, all projections $\hat{N}(p_j)$ of $N(p_j)$ onto their Γ_j are oriented consistently, even if they overlap with several successive neighbors. Thus we can simply compute the centroid of all intersection points as follows:

x_{ij} is the intersection of the line $(p_i, \overrightarrow{c_i p_i})$ with $a_j(t_0', t_1')$ in the half-space of $N(p_j)$ containing p_i. In case of two solutions for the line-circle-intersection, we choose the point closer to p_i. Then $p_i' = \sum_{p_i \in N(p_j), \exists x_{ij}} \frac{x_{ij}}{|X_{ij}|}$ (see Figure 6 for an example). Note that this introduces new point locations as the remaining samples used as vertices are moved, and that we do not blend sharp corners (the points detected as sharp corners and their neighbors) as we want to preserve these features.
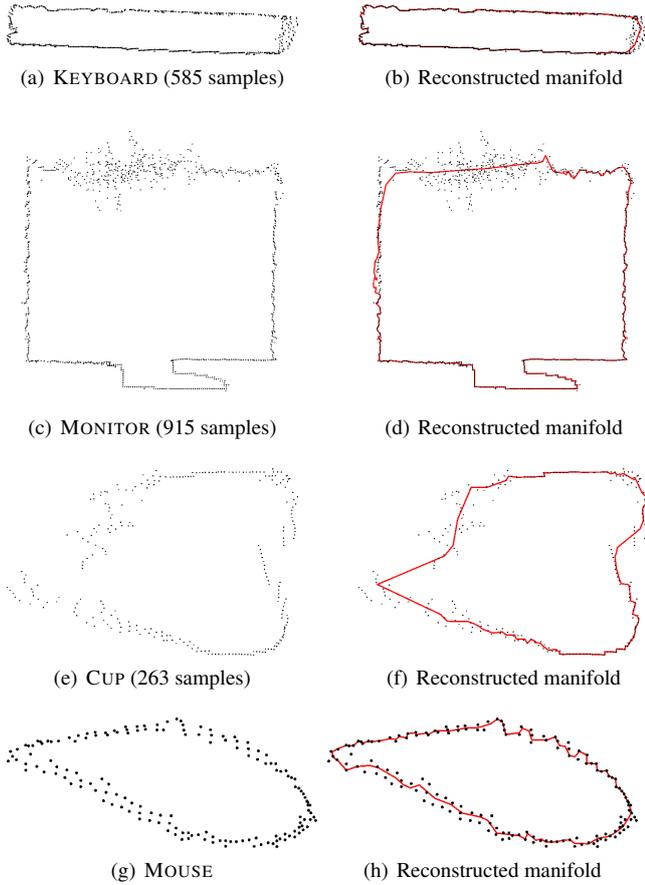
(a) KEYBOARD (585 samples)

(b) Reconstructed manifold

(c) MONITOR (915 samples)

(d) Reconstructed manifold

(e) CUP (263 samples)

(f) Reconstructed manifold

(g) MOUSE

(h) Reconstructed manifold

Figure 7: *Left: Samples representing noisy silhouettes from depth images. Right: Reconstructed single closed manifold curves.*

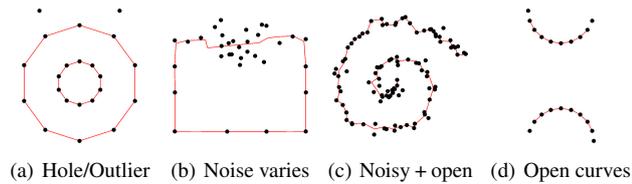(a) Hole/Outlier (b) Noise varies (c) Noisy + open (d) Open curves

Figure 8: *a) Multiply connected curve (object with hole), outliers are not connected. b) Variable high noise: Blended* RECTANGLE. *c) Noisy open curve. d) Two open curves, they do not get connected.*

5. Results

We have run our algorithm successfully on a large number and wide variety of point sets to ensure its robustness: e.g. segmented silhouettes of noisy sensed data [BBP16] (Figure 7), but also contrived shapes representing multiply connected open/closed curves, also corrupted with high noise and outliers (Figures 8, 9, 10, 11 and 14). Additionally, we show its improvements on prior work. We provide open source code for this algorithm that reproduces all result figures and tables of this paper. https://github.com/stefango74/fitconnect.

Comparisons with related work We are aware of only two algorithms in related work which are able to practically reconstruct manifold curves from noisy point sets, *Robust HPR* [MTSM10] and the method of Lee [Lee00], which is limited to open curves. Since we cannot run tests with these algorithms, we refer to the papers for figures for the following comparisons. We show the successful closed manifold reconstruction of the original curve for all the point sets which Robust HPR fails to close (Figure 6, center column, [MTSM10], manages to reconstruct a closed curve only for the APPLE figure), in Figure 10. The following comparison (Figure 11, center column, [MTSM10]) implicitly demonstrates the superiority of our algorithm to *Crust* [ABE98], *CC-Crust* [DMR99] and *Gathan* [DW02], which reconstruct a collection of small unconnected curves instead of the expected closed manifold. Figure 11ab shows our reconstructed close manifold curves for two point sets corrupted by uniformly distributed high noise [Lee00], which are designed for their restriction to open curves. Note that our algorithm closes these holes because they are smaller than nearby features as to our condition in Subsection 4.2. Furthermore, our algorithm handles features of varying size with similarly distributed but adaptive noise ($\delta = \frac{1}{3}$lfs, see Figure 11c). Finally, Figure 13 shows that reconstruction with *Optimal Transport* [DGC-SAD11] is extremely sensitive to the number of iterations. It does not reconstruct a manifold output with any number of iterations for these point sets. The reason why this algorithm fails where ours succeeds (see 14a) is that it starts out with the edge set of a Delaunay triangulation and removes edges iteratively but does not enforce the vertices to be manifold. This method [WYZ*14] uses a similar approach as [DGCSAD11], and we therefore expect it to fail as well for the above-mentioned point sets, as it fails for varying sampling densities (see also Fig. 28 in that paper comparing the two algorithms).

Noise tolerance We have also investigated for which noise extent our algorithm can still successfully reconstruct the topology and features of the original curve. For simulating noise, we use this noise model [MTSM10], which perturbs the original points by a uniform variable in the range $[0, \delta]$ along a unit vector of uniformly chosen random direction. Figure 9 shows that the topology of the circle is reconstructed for noise of δ up to 100% of its radius.

As a more generic case, we have tested the BUNNY from Figure 14 with noise of extent $\delta = \frac{1}{3}$lfs and successfully reconstructed closed manifolds for the 95 samplings with $\varepsilon = 0.005 - 0.1$ in steps of 0.001 (with number of samples varying from 407 to 2491).

In case of noise extent in absolute terms, small features (close curve segments) are omitted from the final reconstruction (see Figure 12 for a more complicated example).
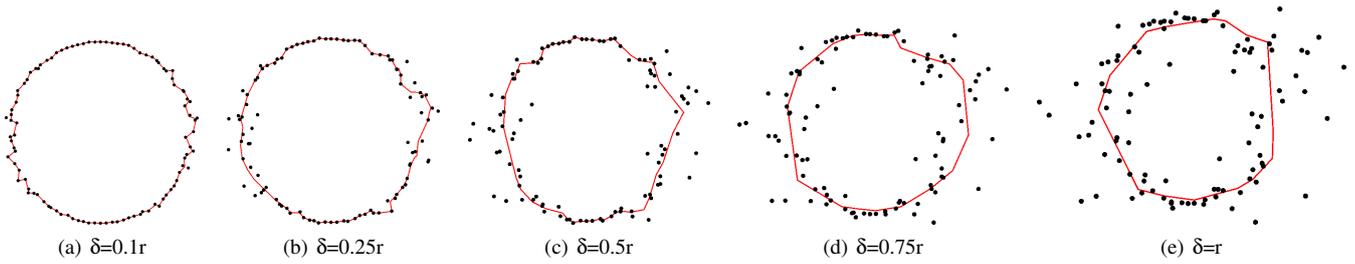
(a) δ=0.1r (b) δ=0.25r (c) δ=0.5r (d) δ=0.75r (e) δ=r

Figure 9: *100 samples on a circle, perturbed with varying (sides: full, top/bottom: zero) noise extent up to δ of its radius.*

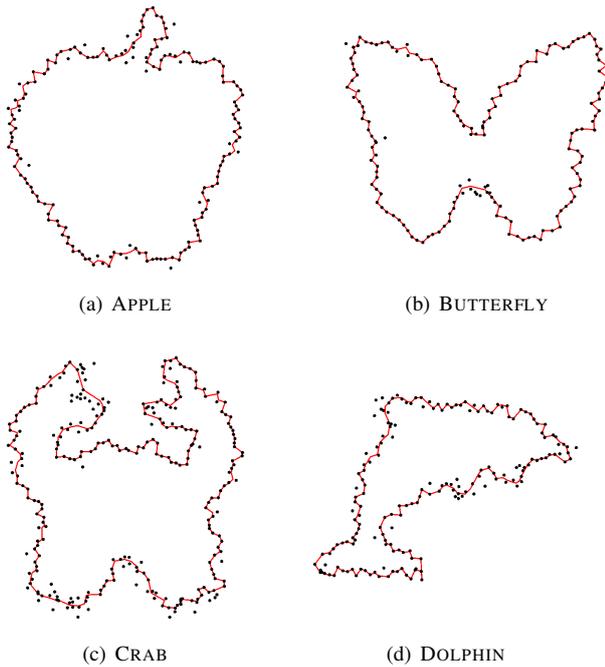(a) APPLE (b) BUTTERFLY

(c) CRAB (d) DOLPHIN

Figure 10: *Successful single manifold reconstruction of point sets [MTSM10] which Robust HPR – except APPLE – and other algorithms fail to connect.*
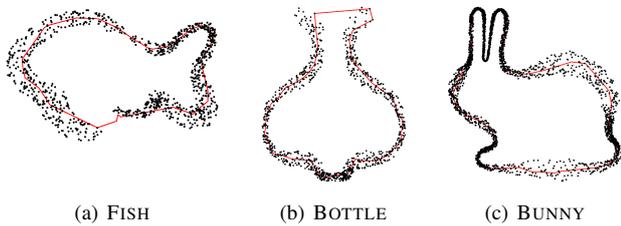

(a) FISH (b) BOTTLE (c) BUNNY

Figure 11: *Left and center: Manifold reconstruction of high-noise point clouds [Lee00]. Right: Bunny with similar noise distribution ($\delta = \frac{1}{3}$ lfs), that additionally reconstructs fine features.*
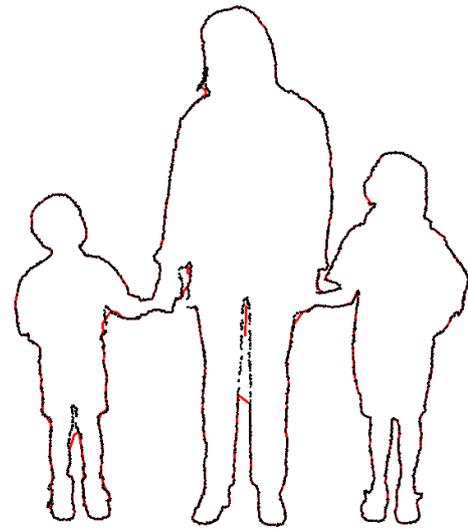

Figure 12: *Noisy silhouette with 3330 points.*

Noise as function of the local feature size In Figure 14 we test increasing noise extent up to 0.5 lfs, for a sampling condition of $\rho = 0.43$ (equivalent to $\varepsilon = 0.3$) as demonstrated in [OMW16]. In Figure 15 we further experiment by sampling a curve with varying densities of a point set while keeping the noise extent fixed at $\frac{1}{3}$ lfs. Our algorithm manages to reconstruct the curve for all the sampling densities.

Detecting local noise extent Since the circular fits of adjacent consistent neighborhoods approximate the original curve locally, we can use the distance of the samples to that fit to estimate the local noise according to our sampling condition: The sample in the neighborhood with maximum distance shows the maximum extent of the noise (see Figure 16), while mean and variance can be extracted from the entire set of samples. These metrics can be put in relation with the radius of the circular fit, in order to bound the error in terms of the local feature size.

Features emerging over noise extent In another experiment we test whether our algorithm is able to reconstruct features emerging over the noise extent. Figure 17 shows a sine wave with fre-
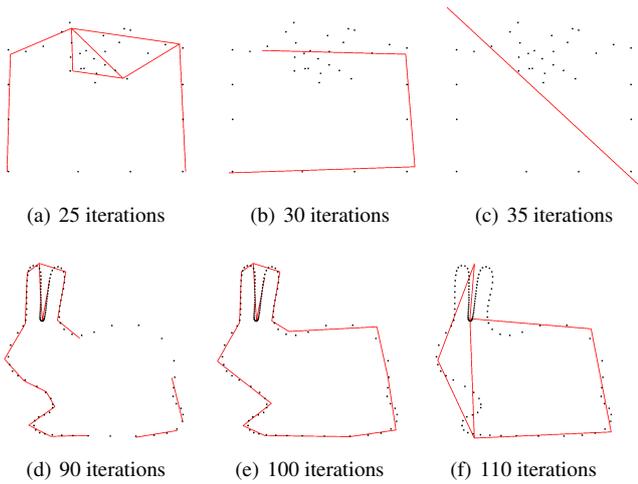
(a) 25 iterations (b) 30 iterations (c) 35 iterations

(d) 90 iterations (e) 100 iterations (f) 110 iterations

Figure 13: *Reconstruction results with* Optimal Transport *[DGC-SAD11] depend strongly on the specified iteration count. Top row: Locally high noise (compare our result in Figure 8b). Bottom row: Non-noisy non-uniform sampling (our result in Figure 14a).*
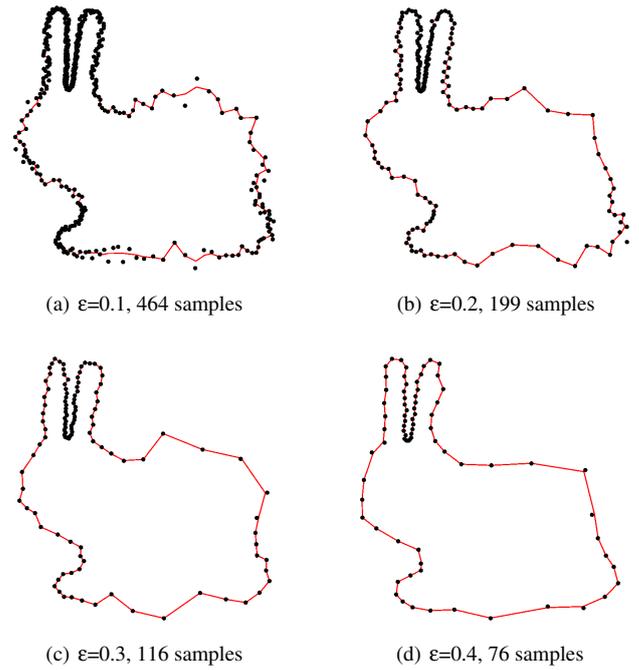

(a) ε=0.1, 464 samples (b) ε=0.2, 199 samples

(c) ε=0.3, 116 samples (d) ε=0.4, 76 samples

Figure 15: BUNNY *sampled with varying ε and perturbed by a noise extent of $\frac{1}{3}$lfs.*

(a) δ=0 (b) δ=0.1lfs
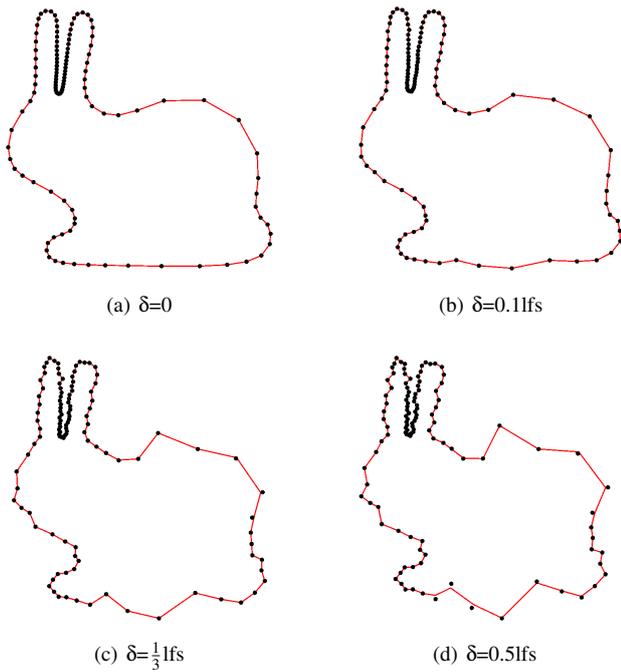

(c) δ=$\frac{1}{3}$lfs (d) δ=0.5lfs

Figure 14: *116 samples on* BUNNY, *sampled with $\rho = 0.43$ (equivalent to $\varepsilon = 0.3$) [OMW16] and perturbed by a noise extent of δ lfs.*
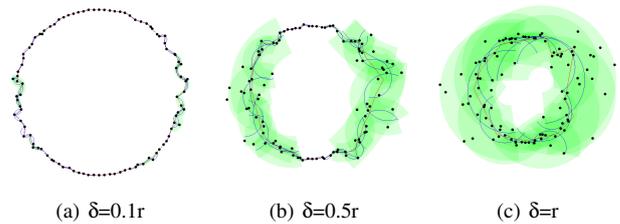

(a) δ=0.1r (b) δ=0.5r (c) δ=r

Figure 16: *100 samples on a circle, perturbed with varying (sides: full, top/bottom: zero) noise extent up to δ of its radius. The arcs represent the local circular fits and the green shaded areas their respective maximum noise extent (not shaded = no noise detected).*

(a) Sine wave with increasing amplitude up to 0.1 of x-extent

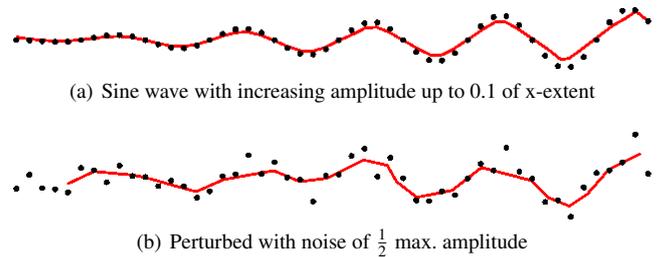(b) Perturbed with noise of $\frac{1}{2}$ max. amplitude

Figure 17: *The features (peaks) of the sine wave exhibit linearly increasing amplitudes in $[0.01..0.1]$ (of x-extent). When the samples are perturbed by noise of $\delta = 0.05$, features are still reconstructed for all amplitudes ≥ 0.02, significantly smaller than the extent of noise which is 250% w.r.t. the feature with amplitude 0.02.*

quency=5 and linearly increasing amplitudes between 0 and 0.1 (of x-extent of the point set). Perturbing the samples with noise of $\delta = 0.05$ permits reconstructing all features with amplitude ≥ 0.02 in this case, that is an extent even below the average vertical displacement (0.025) of the samples, or 250% of the feature size.

Figure/Noise	Input	Output	# Iter	Operations	Complexity	Runtime
CIRCLE 0.1r	100	95	3	440	1078	0.009
CIRCLE 0.25r	100	74	18	2008	2839	0.033
CIRCLE 0.5r	100	55	25	4247	4368	0.056
CIRCLE 0.75r	100	19	51	12450	10291	0.097
CIRCLE 1r	100	16	64	19268	11132	0.122
BUNNY 0	116	116	1	348	1044	0.007
BUNNY 0.1lfs	116	116	1	348	1044	0.007
BUNNY $\frac{1}{3}$lfs	116	116	2	368	1079	0.008
BUNNY 0.5lfs	116	115	2	388	1098	0.008
KEYBOARD	585	448	96	40196	27727	1.168
MONITOR	915	460	38	1191322	418198	13.523
CUP	263	136	304	131532	22660	0.871
MOUSE	157	103	2	3787	4012	0.076
APPLE	170	156	49	2576	2115	0.048
BUTTERFLY	164	154	14	1147	2057	0.028
CRAB	284	222	47	7777	8299	0.220
DOLPHIN	179	146	18	3092	3976	0.070
FISH	1000	46	461	1441983	677138	14.045
BOTTLE	1000	62	462	726360	314934	10.266
BUNNY	2512	263	261	1183601	626739	57.914

Table 1: *We show how our algorithm simplifies the number of samples (input, to output), its time complexity by the number of iterations, and the number of operations on points compared with expected complexity (each circular fit of size M counts as M operations). Runtime in seconds.*

Noise	max. In	mean In	RMS In	max. Out	mean Out	RMS Out
$\delta = 0.1$	0.076	0.016	0.023	0.073	0.013	0.020
$\delta = 0.25$	0.183	0.039	0.059	0.109	0.024	0.034
$\delta = 0.5$	0.367	0.079	0.117	0.126	0.041	0.053
$\delta = 0.75$	0.553	0.118	0.175	0.188	0.053	0.069
$\delta = 1$	0.741	0.155	0.230	0.233	0.079	0.098

Table 2: *Comparison of input (noisy samples) and output (polygon) Hausdorff distance from original circle, for varying noise as shown in Figure 9a-e. All values are in terms of the circle radius.*

Outliers Figure 8a shows some outlier points beside the reconstructed curve. These points are correctly not connected to the curve since they do not fulfill the consistency condition: while their neighbors lie on the curve, those points do not have the outliers reciprocally as neighbors, instead they are consistent with other points on the curve.

Quantitative analysis of reconstruction error Table 2 shows that our simple blending of local neighborhood fits approximates the original curve quite well. All error metrics (maximum, mean and root mean square error) are typically reduced by half or more for the noisier sample sets. The reconstructed curve lies roughly within one third of the noise extent from the original curve (maximum output vs. max. input error). We also determined the Hausdorff distance of the reconstructed curve w.r.t. the noisy points from the original curve. For 100 unit circles tested with $\delta = 0.25$, the overall maximum error for the input is 0.249, for the output, 0.242, and the maximum ratio between output and input per circle is 1.002. The distance of the reconstructed curve from the original therefore seems to be limited in practice by the extent of the noise.

Time complexity and convergence The number of effected operations matches the expected complexity $O(\sum_{p_i}^{P} |N(p_i)|^2)$ for the

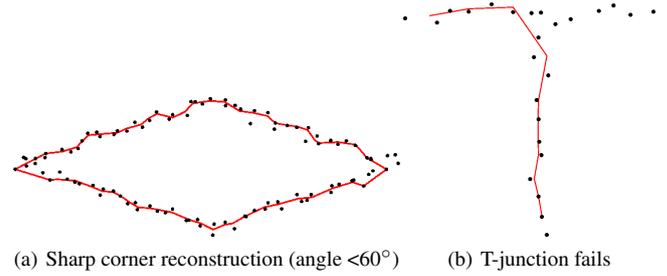(a) Sharp corner reconstruction (angle <60°)   (b) T-junction fails

Figure 18: *Handling of sharp corners and T-junctions.*

output point set P and their fitted neighborhood sizes $N(p_i), p_i \in P$ roughly in the order of magnitude (see Table 1). Handling large noisy clusters or boundaries therefore increases the runtime of our non-optimized algorithm significantly. Algorithm 1 shows that points will only ever be eliminated, never added from the original point set of size N. Since the neighborhood of a point can grow at most to size N, it would be marked as handled eventually as well in that case, so the algorithm will always converge to a solution in polynomial time. The numbers for the BUNNY sample sets let us suspect that the complexity is linear as long as the extent of the noise stays within 0.5 lfs. Robust HPR [MTSM10] takes 1–2 seconds (for small point sets with little noise) while DeGoes et al. [DGCSAD11] show timings similar to ours (range in seconds up to a minute).

Limitations As shown in Figures 14, 15 and 16, our method manages to reconstruct very strong noise up to the extent of the local feature size. However, the less densely a feature is sampled, and the more the local feature size varies between close curves, the closer their noisy samples become, which might connect these samples and thus merge their features. Limiting the noise extent to $\frac{1}{3}$ of the local feature size seems to work well for the general case (see Figure 15). Figure 10c shows that features which do not emerge over the noise extent are oversmoothed (indentation of left pince).

T-junctions (see Figure 18b) do not represent a manifold curve, and therefore our fitting operator does not handle these. Open curves may not always be reconstructed to the farthest point, as for example in Figure 8c, if our algorithm marks these points as redundant, such as in Figures 17, 18. However, this could easily be fixed by a post-processing step.

6. Conclusion and Future Work

We have presented an algorithm that solves the extensively studied problem of reconstructing simple curves from arbitrarily noisy points, with applications e.g. in recovering silhouettes of 3D objects in sensed data, but most importantly, providing groundwork for reconstructing surfaces from highly noisy 3D data. Our reconstruction extends seamlessly from an existing algorithm, HNN-CRUST, to handle samples polluted by high noise extents. Additionally, it simplifies the output curve without losing features and denoises it. The reconstructed curve is guaranteed to fulfill two conditions w.r.t to the input points and analysis shows that it stays within the same

distance to the original curve as the error of the noisy points. The algorithm runs in reasonable time to be of practical use, with runtime depending on the extent of noisy clusters in the data. The convergence to a robust solution can be verified using the open source available online.

Additionally to the boundary and sharp-corner detectors, we are considering to extend the sharp-corner detector to T-junctions and intersections, by considering more than two open curves in the neighborhood disk of points. We also plan to incorporate statistical noise models into the circular fits, e.g., with sensor-specific properties, to improve our denoising post-process because known noise extents also permit to remove smaller extents which are not detected by our algorithm. The consistent ordering along a manifold guaranteed by our two conditions enables anisotropic denoising as the curve is locally planar and can be deformed in function of the neighborhood points. However, we currently work on extending the algorithm to reconstructing surfaces in 3D, as we consider this to be its most exciting potential.

7. Acknowledgments

References

[ABE98] AMENTA N., BERN M. W., EPPSTEIN D.: The crust and the beta-skeleton: Combinatorial curve reconstruction. *Graphical Models and Image Processing 60*, 2 (1998), 125–135. 2, 7

[ACSTD07] ALLIEZ P., COHEN-STEINER D., TONG Y., DESBRUN M.: Voronoi-based variational reconstruction of unoriented point sets. In *Symposium on Geometry processing* (2007), vol. 7, pp. 39–48. 2

[Alt01] ALTHAUS E.: *Curve Reconstruction and the Traveling Salesman Problem*. Doctoral dissertation, Universität des Saarlandes, 2001. 2

[ASC*09] AL-SHARADQAH A., CHERNOV N., ET AL.: Error analysis for circle fitting algorithms. *Electronic Journal of Statistics 3* (2009), 886–911. 4

[Att97] ATTALI D.: r-regular shape reconstruction from unorganized points. In *Symp. on Computational Geometry* (1997), pp. 248–253. 2

[BBP16] BIRKAS D., BIRKAS K., POPA T.: A mobile system for scene monitoring and object retrieval. In *Proceedings of the 29th International Conference on Computer Animation and Social Agents* (2016), ACM, pp. 83–88. 2, 7

[CFG*05] CHENG S.-W., FUNKE S., GOLIN M., KUMAR P., POON S.-H., RAMOS E.: Curve reconstruction from noisy samples. *Computational Geometry 31*, 1-2 (2005), 63–100. 2

[CL05] CHERNOV N., LESORT C.: Least squares fitting of circles. *Journal of Mathematical Imaging and Vision 23*, 3 (2005), 239–252. 4

[DG06] DEY T. K., GOSWAMI S.: Provable surface reconstruction from noisy samples. *Computational Geometry 35*, 1-2 (2006), 124–141. 2

[DGCSAD11] DE GOES F., COHEN-STEINER D., ALLIEZ P., DESBRUN M.: An optimal transport approach to robust reconstruction and simplification of 2d shapes. In *Computer Graphics Forum* (2011), vol. 30, Wiley Online Library, pp. 1593–1602. 2, 7, 9, 10

[DK99] DEY T. K., KUMAR P.: A simple provable algorithm for curve reconstruction. In *Proc. 10th ACM-SIAM SODA '99* (1999), pp. 893–894. 2

[DMR99] DEY T. K., MEHLHORN K., RAMOS E. A.: Curve reconstruction: Connecting dots with good reason. *In Proc. 15th ACM Symp. Comp. Geom 15* (1999), 229–244. 2, 7

[DT14] DUARTE P., TORRES M. J.: Smoothness of boundaries of regular sets. *Journal of mathematical imaging and vision* (2014), 1–8. 2

[DT15] DUARTE P., TORRES M. J.: r-regularity. *Journal of Mathematical Imaging and Vision 51*, 3 (2015), 451–464. 2

[DW02] DEY T. K., WENGER R.: Fast reconstruction of curves with sharp corners. *Int. J. Comp. Geom. Appl. 12*, 5 (2002), 353 – 400. 2, 7

[EKS83] EDELSBRUNNER H., KIRKPATRICK D. G., SEIDEL R.: On the shape of a set of points in the plane. *IEEE Trans. Inf. Theor. IT-29*, 4 (1983), 551–559. 2

[FJ11] FEISZLI M., JONES P. W.: Curve denoising by multiscale singularity detection and geometric shrinkage. *Applied and Computational Harmonic Analysis 31*, 3 (2011), 392–409. 3

[FMG94] FIGUEIREDO L. H. D., MIRANDAS GOMES J. D.: Computational morphology of curves. *Vis. Comp. 11*, 2 (1994), 105–112. 2

[FR01] FUNKE S., RAMOS E. A.: Reconstructing a collection of curves with corners and endpoints. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms* (Philadelphia, PA, USA, 2001), SODA '01, Society for Industrial and Applied Math., pp. 344–353. 2

[GG07] GUENNEBAUD G., GROSS M.: Algebraic point set surfaces. In *ACM Transactions on Graphics (TOG)* (2007), vol. 26, ACM, p. 23. 4

[Gol99] GOLD C.: Crust and anti-crust: a one-step boundary and skeleton extraction algorithm. In *Proc. of the 15th ann. Symp. on Computational geometry* (New York, NY, USA, 1999), SCG '99, ACM, pp. 189–196. 2

[KH13] KAZHDAN M., HOPPE H.: Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG) 32*, 3 (2013), 29. 2

[KR85] KIRKPATRICK D. G., RADKE J. D.: A framework for computational morphology. *Computational Geometry* (1985), 217–248. 2

[Lee00] LEE I.-K.: Curve reconstruction from unorganized points. *Computer aided geometric design 17*, 2 (2000), 161–177. 2, 7, 8

[Len06] LENZ T.: How to sample and reconstruct curves with unusual features. In *Proceedings of the 22nd European Workshop on Computational Geometry (EWCG)* (Delphi, Greece, March 2006). 2

[MTSM10] MEHRA R., TRIPATHI P., SHEFFER A., MITRA N. J.: Visibility of noisy point cloud data. *Computers & Graphics 34*, 3 (2010), 219–230. 2, 7, 8, 10

[OM13] OHRHALLINGER S., MUDUR S.: An efficient algorithm for determining an aesthetic shape connecting unorganized 2d points. In *Comp. Graph. Forum* (2013), vol. 32, Wiley Online Library, pp. 72–88. 2

[OMW16] OHRHALLINGER S., MITCHELL S. A., WIMMER M.: Curve reconstruction with many fewer samples. In *Computer Graphics Forum* (2016), vol. 35, Wiley Online Library, pp. 167–176. 2, 3, 8, 9

[PM16] PARAKKAT A. D., MUTHUGANAPATHY R.: Crawl through neighbors: A simple curve reconstruction algorithm. In *Computer Graphics Forum* (2016), vol. 35, Wiley Onl. Library, pp. 177–186. 2

[ST09] STELLDINGER P., TCHERNIAVSKI L.: Provably correct reconstruction of surfaces from sparse noisy samples. *Pattern Recognition 42*, 8 (2009), 1650–1659. 2

[Ste08] STELLDINGER P.: Topologically correct surface reconstruction using alpha shapes and relations to ball-pivoting. In *Pattern Recognition, 2008. ICPR 2008. 19th Int'l Conference on* (2008), IEEE, pp. 1–4. 2

[WYZ*14] WANG J., YU Z., ZHANG W., WEI M., TAN C., DAI N., ZHANG X.: Robust reconstruction of 2d curves from scattered noisy point data. *Computer-Aided Design 50* (2014), 27–40. 2, 7