Eurographics Symposium on Geometry Processing 2011
Mario Botsch and Scott Schaefer
(Guest Editors)

Volume 30 (2011), Number 5

On the shape of a set of points and lines in the plane

M. van Kreveld[1], T. van Lankveld[1], and R. C. Veltkamp[1]

[1]Department of Information and Computing Sciences, Utrecht University, The Netherlands

Abstract

Detailed geometric models of the real world are in increasing demand. LiDAR data is appropriate to reconstruct urban models. In urban scenes, the individual surfaces can be reconstructed and connected to form the scene geometry. There are various methods for reconstructing the free-form shape of a point sample on a single surface. However, these methods do not take the context of the surface into account. We present the guided α-shape: an extension of the well known α-shape that uses lines (guides) to indicate preferred locations for the boundary of the shape. The guided α-shape uses (parts of) these lines as boundary where the points suggest that this is appropriate. We prove that the guided α-shape can be constructed in $O((n+m)\log(n+m))$ time, from an input of n points and m guides. We apply guided α-shapes to urban reconstruction from LiDAR, where neighboring surfaces can be connected conveniently along their intersection lines into adjacent surfaces of a 3D model. We analyze guided α-shapes of both synthetic and real data and show they are consistently better than α-shapes for this application.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.5]: Computational Geometry and Object Modeling—Boundary representations; Computer Graphics [I.3.5]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

1. Introduction

One of the major areas of interest in geometry processing is reconstructing geometry from point clouds. The quality of reconstruction methods depends heavily on the restrictions and goals of their application. General reconstruction methods aim at a wide range of applications, but often lack the quality of methods aimed at a specific application. An application that has recently received much interest is urban reconstruction, in which most surfaces are planar. We present a LiDAR-based urban reconstruction approach and, specifically, we present a novel method that constructs the boundary, the guided α-shape, of a planar point set to achieve that neighboring surfaces can be connected along shared boundary edges. The guided α-shape is similar to the α-shape that is guided by lines that indicate appropriate locations for the shape boundary, see Figure 4.

We first present related work in Subsection 1.1 and we specifically go into more detail on the α-shape in Section 2. In Section 3 we define the guided α-shape. We prove that the guided α-shape can be constructed as efficiently as the α-shape in Subsection 3.4. We compare the quality of the resulting shape to the α-shape in Section 4. Finally, we conclude the paper and present directions for further research in Section 5.

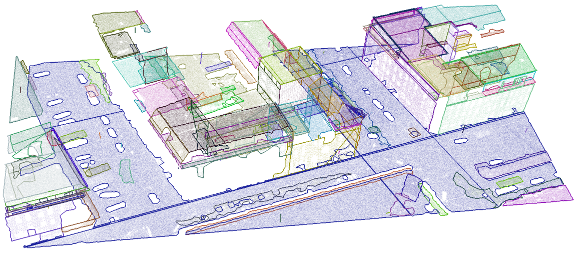


Figure 1: *The guided α-shapes in an urban LiDAR scene. These connect neighboring surfaces along an edge.*

1.1. Related Work

Many methods for reconstructing the geometry of a point cloud try to fit a curved surface or manifold to all the data points [ACSTD07, TTC07]. These methods usually assume that the points are regularly distributed over the boundary of an object. If the scene contains many flat surfaces, as in many urban scenes, these methods are at a disadvantage. In this case we can assume the measurements can be partitioned into subsets that are each approximated by a planar polygon. The problem of reconstructing the scene geometry can then be reduced to determining the supporting planes and the boundaries of the polygons that make up the scene. Manifold methods usually disregard this assumption of planarity and construct a triangular mesh plagued by erratic regions caused by noise.

Other methods take into account the piecewise nature of the scene [LWY08, SYM10]. However, in this case identifying the sharp features becomes the main problem. Extracting these sharp features from a collection of planes, or other simple surfaces, is almost trivial.

Geodesics research has yielded various methods for reconstructing geographic scenes from LiDAR data [Rot03, VGSR04, YHNF03, ZN08]. These methods exploit the abundance of planar surfaces, but they assume a regular grid in the horizontal plane containing one height measurement per cell. This leads to two problems: they need complicated procedures to compensate for empty grid cells, and they are unable to handle multiple measurements per grid cell. For example, they are unable to reconstruct the geometry under overhanging structures or vegetation.

Various methods can be used to cluster a point cloud into surfaces, like RANSAC [SWK07] and region growing [TTC07]. Schnabel *et al.* [SDK09] use such surfaces to reconstruct the object, using a version of marching cubes [KBSS01]. This greatly reduces the simplicity of the planar surfaces, while we wish to retain their conciseness. The remaining problem is to bound each surface. We do this by reconstructing the shape of the points in each surface. This shape reconstruction process can usually be guided by the intersection lines of the surface with its neighbors.

Multiple methods have been proposed to find a region containing a point set. Structures like the crust, β-skeleton and γ-neighborhood graph [ABE98, Vel92] assume the points were sampled from the boundary of the region. If the points are evenly distributed over the interior of the shape, the most used structure is the α-shape [CGPZ05, EKS83]. More complex methods can produce the shape of non-evenly distributed point sets [CGPZ05, TC98].

A problem that has not been addressed previously is computing the shape when indications are given for the location of parts of the boundary. We present the guided α-shape: a novel extension of the α-shape that is influenced by lines or line segments. These shapes are more appropriate for urban geometry reconstruction, because in urban scenes the shapes should be connected along shared boundary segments. We present a method to construct the guided α-shape, and we compare our method in efficiency and applicability to the α-shape. Finally, we present some results on applying our method to urban reconstruction.

Similar to the way α-shape construction is based on the Delaunay triangulation and Voronoi diagram, our method is based on the constrained Delaunay triangulation [Che89] and generalized Voronoi diagrams [Kir79]. The Delaunay empty-circle criterion and Voronoi closest-point criterion are natural ways of determining the interior of a shape of a point set. Unlike the standard α-shape that is constructed purely from the data points, the guided α-shape is also guided by lines and therefore constructed using structures that are influenced by lines.

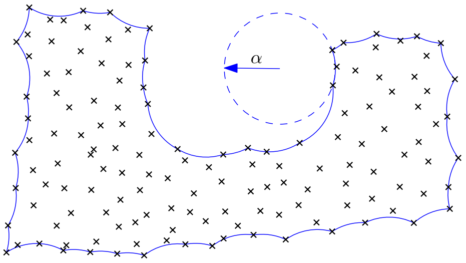


Figure 2: *The α-hull. Its complement contains all empty α-disks. The α-shape is the straight line version of the hull.*

2. The shape of a set of points

A ‘shape’ is the outer form or appearance of an object; it is the outline of the region that makes up the object. If a shape is unknown, it can be reconstructed from a point sample of the shape’s interior. Few papers explicitly define the shape of a set of points in the plane, but there are some considerations on the intuitive concept of this shape. We assume a shape is bounded by straight line segments. We do not require the shape to be connected, but points that are close together are in the same component of the shape. Conversely, a component of the shape cannot contain a large region void of points. What constitutes ‘close’ and a ‘large region’ depends both on the distribution of the points and the abstraction level at which the shape is viewed. Therefore, the shape of a point set is linked to some scale parameter; when the scale of the shape increases, the amount of detail in the shape decreases.

Edelsbrunner *et al.* [EKS83] introduced α-shapes as the shape of a point set at a certain level of detail, α. The α-shape is the straight line dual of the difference between the whole plane and all empty open disks with radius α, as shown in Figure 2. For completeness, the α-shape definition is repeated here in condensed form; details on its application as well as the proofs of Observation 1 can be found in [EKS83].

Definition 1 (α-shape) *Given a point set S, a point $p \in S$ is α-extreme if there exists an empty open disk (i.e., not containing any point from S) of radius α with p on its boundary. Two points $p, q \in S$ are called α-neighbors if they share such an empty disk. The α-shape of S is the straight line graph whose vertices are the α-extreme points and whose edges connect the respective α-neighbors.*

Observation 1 *The α-shape of a point set S partitions the plane into interior and exterior faces. The interior faces do not contain the center of an empty open disk with radius α. The interior faces of the α-shape are exactly the union of all triangles of a Delaunay triangulation with a circumscribed circle with radius $\leq \alpha$.*

3. Guided α-shape

The α-shape cannot take into account the context of the shape. In surface reconstruction, we may know some of the neighboring surfaces, giving an indication of where the surface ends. The guided α-shape takes advantage of these indicators to construct a better shape.

The guided α-shape is the shape of a set of points in the plane, which we will refer to as sites, where a set of non-intersecting line segments, called guides, indicate preferred locations for parts of the boundary. Guides determine the boundary of the shape only near sites. Further away from guides or sites, the shape is the same as the α-shape.

The definition of the guided α-shape is based on two concepts: connected α-disks and α-projections. A connected α-disk is a connected component of a disk. This structure replaces the α-disk in determining the guided α-shape as compared to the original α-shape. This limits the influence of sites across a guide.

Sites can give α-projections on guides, if these conform to some conditions, listed in Definition 3. These conditions are chosen to elegantly identify appropriate parts of the guides for the shape boundary. The α-projections provide additional points, similar to Steiner points added in conforming Delaunay triangulations [ET92, Rup95]. However, unlike Steiner points, projections lie infinitesimally close to a guide and strictly to one side of it. As a result, the α-projections are the vertices of the shape boundary where the point set is near a guide.

Definition 2 (Connected α-disk) *Given a site or projection p, a set of guides G and a positive value α, let D denote an open disk of radius α with p on its boundary. A connected α-disk of p is the closure of a connected region of $D \setminus G$ that contains p, minus the boundary of D (shown in Figure 3).*

Definition 3 (α-projections) *Given a set of sites S, a set of non-intersecting guides G, and a positive real value α, $P^{\alpha}_{S,G}$ is a set of points near G, called α-projections. Each α-projection $p \in P^{\alpha}_{S,G}$ has a site $s \in S$, called the projection's source, such that the following three conditions hold.*

> ***Closest:*** *there is a $g \in G$ such that if q is the point on g closest to s, then p is infinitesimally close to q and on the same side of g as s.*
> ***Visible:*** *the line segment connecting s and p does not intersect any guide in G.*
> ***Neighbor:*** *p and s share a connected x-disk empty of sites, where $x \leq \alpha$.*

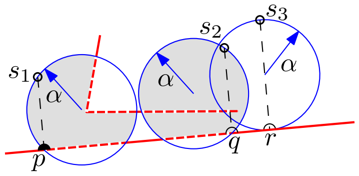


Figure 3: *The gray regions are connected α-disks; they contain the dashed parts of the guides, but not the circular blue boundaries. p is an α-projection of s_1; r and q violate one of the conditions in Definition 3, so they are not projections.*

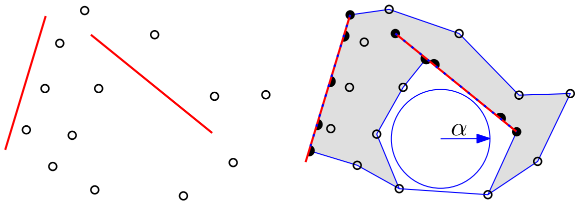


Figure 4: *A set of sites and guides and the line-guided α-shape of this set. The (half) disks are α-projections.*

There are three important subtleties to the α-projections. Firstly, the α-projection p of site s onto guide g is not exactly on g. Therefore, the connected component containing p of an α-disk is always on the same side of g as s. This, in turn, means all connected α-disks of p are on the same side of g as s. This is not the same for a site located on a guide; this site has connected α-disks on either side of the guide.

Secondly, an α-projection p of site s always has an empty connected α-disk. Imagine taking a sequence of connected α-disks of p, each next one further away from s than the current. At some point, the connected α-disk will be empty.

Finally, the point q on a guide closest to site s may be the guide endpoint. There is an α-projection of s near q if it follows the conditions of Definition 3. Multiple sites may project onto this endpoint, giving multiple α-projections near q. Again, all connected α-disks of these projections are on the same side of g.

Armed with the definitions for the connected α-disk and projections, we can now define the line-guided α-shape, shown in Figure 4. In the absence of guides, this shape is the same as the original α-shape; near guides, the shape is adapted following the shortest route to use the guides as boundaries.

Definition 4 (Line-guided α-shape) *Given a set of sites S, a set of non-intersecting guides G, and a positive real value α, let $P^{\alpha}_{S,G}$ denote the set of α-projections of S onto G.*

A point $s \in S \cup P^{\alpha}_{S,G}$ is α-extreme if there exists a connected α-disk of s empty of sites and α-projections. Two points in $S \cup P^{\alpha}_{S,G}$ are α-neighbors if they share a connected α-disk that is empty of sites and α-projections.

The line-guided α-shape of S and G is the straight line graph whose vertices are the α-extreme points and whose edges connect the respective α-neighbors.

Observation 2 *Four observations follow naturally from the terms defined above.*

1. *If there are no guides, the α-extreme candidates are the α-extreme points of the original α-shape.*
2. *All α-extreme sites are also an α-extreme candidate.*
3. *All projections are α-extreme.*
4. *Because a projection must share an α-disk with its source, these can be no further than 2α apart. This also means that each guide has a maximal reach of 2α in which it can influence the shape.*

The line-guided α-shape is like an α-shape that is influenced by guides. However, some situations produce undesirable artifacts in line-guided α-shapes, as shown in Subsection 3.1. This problem is solved by enforcing an additional constraint on the connected α-disks, leading to the definition of guided α-shapes. As an additional advantage, this constraint drastically reduces the number of projections, as explained in Subsection 3.2. In Subsections 3.2 to 3.5, we present our algorithm to construct the guided α-shape and we discuss some artifacts arising in practice. The missing proofs can be found in [vKvLV11].

3.1. The influence of endpoints

The primary goal of the guided α-shape is to adjust the α-shape such that it uses the supplied guides as boundary where appropriate. The definitions given in the previous section partially achieve this goal: the guides influence the shape if they are near it. However, in some practical cases, the influence of guides is not appropriate, see Figure 5.

The line-guided α-shape can be improved by taking into account the locations of the endpoints of the guides. In some cases, the locations where a guide starts and ends is not important, or not exactly known. In other cases there is a clear indicator for the endpoint of a boundary edge. An example is the intersection point where three neighboring surfaces meet.

For the remainder of the paper, we will assume all endpoints of the guides are such explicit indicators of boundary corners. If the endpoint of a guide is less explicit, it can be placed very far away and the orthogonal projection feature of the guided α-shape will make sure that only the part of the guide near the point set is used. To make the endpoints more explicit in guiding the shape, they disrupt empty connected α-disks and provide more projections.

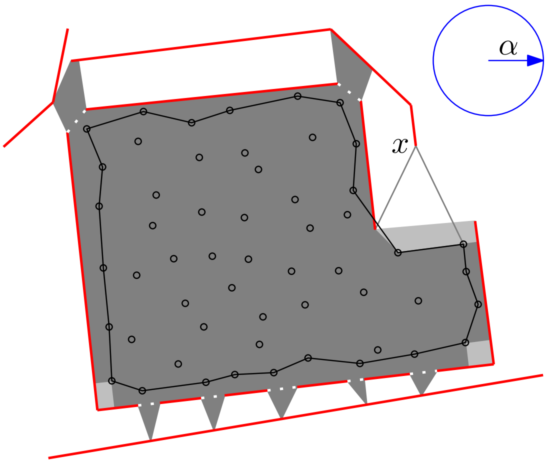


Figure 5: *An example of realistic input where the line-guided α-shape (dark gray) gives undesired results. Slightly adjusting the criteria near guide endpoints gives the guided α-shape. This shape does not cross the small gaps indicated by the dotted lines, it covers the light gray regions, and it does not connect to x. The original α-shape is black.*

Definition 5 (Guided α-shape) *A connected α-disk is valid if it is empty and does not contain a guide endpoint. For α-projections each endpoint of a guide is handled as if it is a degenerate (length 0) guide. The guided α-shape is the line-guided α-shape with all references to empty connected α-disks replaced by valid connected α-disks.*

The guided α-shape follows the guides and their endpoints where they have sites nearby, while reverting to the α-shape in absence of such indicators. The validity constraint makes sure that the shape will not cross small gaps between endpoints, unless there are sites that indicate the shape continues on the other side.

The endpoints also provide additional α-projections, because they are handled as additional guides. If a site is near an endpoint, this endpoint is the point on the additional guide closest to the site. The result is that an α-projection is added in convex corners near sites.

Like the α-shape, the guided α-shape separates interior and exterior faces. Similar to Observation 1, the interior faces are the union of the triangles in the constrained Delaunay triangulation of the sites, guides, and projections with a circumdisk of radius $x < \alpha$. This is proven in [vKvLV11] comparably to the proofs of Observation 1.

3.2. Number of projections

Validity, as described in Definition 5, improves the resulting shape. It also drastically reduces the worst-case bound on the number of projections. Because each site is projected onto any guide at most once, there can be at most $O(n*m)$ projections of n sites onto m guides. It is possible to construct examples that show $\Theta(n*m)$ is indeed the worst-case bound on the number of projections if validity is not enforced. However, we will show that if the connected α-disks must be valid, there are at most $O(n+m)$ projections.

We use two steps to prove there are a linear number of projections if the connected α-disks must be valid. We first prove that in the constrained Delaunay triangulation (CDT) on the sites and guides, each projection must be inside the circumdisk of a triangle incident to its source. We then show that any circumdisk of a CDT triangle can contain at most two projections of each of its incident vertices. Because the CDT has $O(n+m)$ triangles, there are $O(n+m)$ projections. Figure 6 shows most of the structures used in the proofs.

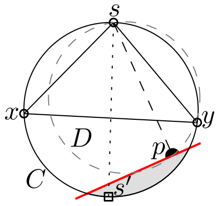


Figure 6: *The different structures used in the proofs of Lemmas 1 and 2.*

Lemma 1 *In the CDT of the sites and guides, each projection is inside the circumdisk of a triangle incident to its source.*

Lemma 2 *Given a set of n sites and m non-intersecting guides, there are $O(n+m)$ projections.*

Proof We will show that each circumdisk C of a CDT triangle incident to a site s is divided into two parts by the diagonal between s and opposite point s'; each half of C can contain at most one projection of s. As there are $O(n+m)$ such circumdisks in the CDT and each is incident to at most three sites, this constitutes a proof. Furthermore, there are at most $2m$ projections on an endpoint of a guide.
Assume a half of disk C contains two projections p_1 and p_2 of s on guides g_1 and g_2 respectively. Both g_1 and g_2 must completely intersect C and we call the regions behind the guides relative to the site r_1 and r_2 respectively. Because the projections are not on an endpoint of g_1 or g_2, the lines connecting source and projection are orthogonal to their respective guide. According to Thales' theorem, both r_1 and r_2 must contain s'. Therefore one of three cases are possible: r_1 contains p_2 and therefore sp_2 intersects g_1, or similarly r_2 contains p_1, or g_1 and g_2 intersect inside C. As all three cases contradict the assumptions, both halves of C can contain at most one projection of s. □

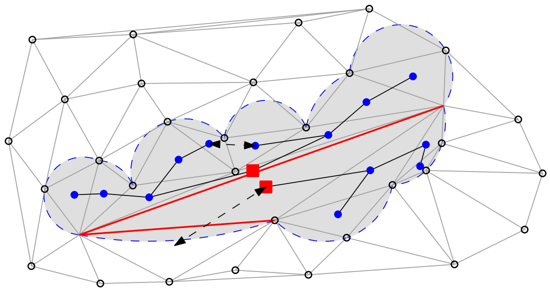


Figure 7: *The cumulus trees of a constraint. The red squares are the roots of the trees and the black edges show which nodes are connected. The nodes connected by the dashed arrows do not share an edge in the cumulus tree, because the intersection of their disks does not overlap the constraint (top), or the edge in the CDT is a constraint (bottom).*

3.3. Cumulus tree

There is an interesting inherent structure in the collection of CDT triangles for which the circumdisks overlap a certain constraint. This structure can be used to both efficiently find the projections and calculate the radius of the smallest valid connected disk they share with their sources, as explained in Subsection 3.4. We call this structure a *cumulus tree*, based on its cloud-like appearance. The two cumulus trees of a constraint are shown in Figure 7

Definition 6 (Cumulus tree) *A cumulus tree $T(c)$ associated with constraint c of a CDT is a subset of the dual graph of the CDT. Each node of $T(c)$ corresponds to a triangle in the CDT for which the circumdisk overlaps c. Two nodes share an edge if their corresponding triangles share a non-constrained edge e in the CDT and the intersection of their circumcircles overlaps c.*

Each constraint c is associated with two cumulus trees $T(c)$, rooted in the two triangles incident to c. These cumulus trees have nice properties, given in the following two lemmas. Lemma 3 implies that the graph of a cumulus tree is actually a tree. Note that a cumulus tree $T(c)$ need not contain all circumcircles in the CDT that overlap c, only those not separated from c by another constraint.

Lemma 3 *Let $T(c)$ be a cumulus tree of a constraint c, let u be the node corresponding to the triangle incident to c, and let $D(x)$ denote the circumdisk of the triangle associated with node x. Then for any simple path on $T(c)$ from u to another node w, we have: if v lies on the path between u and w, then $\{D(w)\cap c\} \subseteq \{D(v)\cap c\}$.*

Proof Because u is incident to c, $\{D(u)\cap c\} = c$. For any edge vw in the cumulus tree, both $D(w)\cap c$ and $D(v)\cap c$ lie on the same side of the supporting line of the edge e shared by the triangles associated with w and v. Let v be the node associated with the triangle on the same side of e as $D(v) \cap c$. We can now parameterize the line segment l between the centers of $D(w)$ and $D(v)$ using parameter $t \in [0,1]$, such that $l(0)$ is the center of $D(w)$ and $l(1)$ is the center of $D(v)$. Using this parameterization, let $D(t)$ be the circumdisk of e centered on $l(t)$. Now for any $m, n \in [0,1]$, $\{D(m) \cap c\} \subseteq \{D(n) \cap c\}$ if and only if $m \leq n$. □

Lemma 4 *Given a CDT on a set of n points and m constraints, the total number of nodes in the cumulus trees of all the constraints is $O(n+m)$.*

3.4. Constructing the guided α-shape

The guided α-shape can be computed in six simple steps. We show that, given n sites, m guides, and a value of α, these steps take $O((n+m)\log(n+m))$ time to determine the guided α-shape. The steps of our method are:

1. Compute the constrained Delaunay triangulation (CDT) of the sites and guides.
2. Compute all possible projections (for any α).
3. Compute the minimal α value for each projection.
4. Select the projections for the given α value.
5. Adjust the CDT to incorporate the projections.
6. Extract the guided α-shape.

The following subsections show the algorithms and running times for steps 2, 3, and 5. The other steps have trivial or well known time bounds:

- Step 1 takes $O((n+m)\log(n+m))$ time and gives an $O(n+m)$ size structure [Che89].
- Step 4 can be done from the collection of all $O(n+m)$ projections in $O(n+m)$ time, by iterating over the projections and keeping only those for which the minimal α value is smaller than the specified α value.
- Step 6 can be done from the adjusted CDT in $O(n+m)$ time, by iterating over all $O(n+m)$ edges and triangles in the structure and keeping those that comply with two conditions. Firstly, they must have a circumcircle with radius smaller than α. Secondly, they cannot be incident to an endpoint of a guide that is not projected onto by a source in the wedge containing the edge or triangle. Note that the first condition is the same as the condition for computing the α-shape from the Delaunay triangulation. The second condition makes sure that only edges on the correct side of a guide are in the shape.

3.4.1. Finding projections

In step 2 of our method, all possible projections are computed. At this stage, α is temporarily set to ∞ to provide the projections; the projections that are too far away will be removed in step 4. The algorithm for computing the projections starts from a triangle incident to a constraint c and traverses the cumulus tree. For each encountered triangle t, its incident sites are projected onto c. The projections in the circumcircle of t are the projections with an empty connected α-disk.

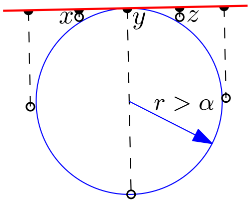


Figure 8: *α-projections have a minimum α value; at smaller values they should not influence the guided α-shape.*

It is not necessary to explicitly construct the cumulus tree beforehand. Each triangle incident to a constraint c must be a root. After handling a triangle t during traversal, each triangle t_a that shares a non-constrained edge with t is checked. If the intersection of the circumdisks of t and t_a contains c, then t and t_a share an edge in the cumulus tree. In this case, t_a is handled. The number of triangles that are checked, but not handled can be no more than twice the number of triangles that are handled. Because each triangle can be handled in $O(1)$ time, it follows from Lemma 1 and 4 that this algorithm constructs all projections in $O(n+m)$ time.

3.4.2. Selecting projections

In step 3 of our method, the minimum α at which each projection appears is calculated. This is the radius $\tilde{\alpha}$ of the smallest valid connected disk shared by the projection and one of its sources. These values can be efficiently computed by using the CDT and its cumulus trees.

Lemma 5 *All the $\tilde{\alpha}$ values at which a projection appears can be computed from the CDT in $O(n+m)$ time.*

The necessity of checking the size of the x-disk is shown in Figure 8. Here, projection y is a valid projection for larger α values, but at smaller α values x and z are α-neighbors and a false projection y should not interfere here.

3.4.3. Triangulated structure

For step 5 of our method, the algorithm is similar to the algorithm for computing the constrained Delaunay triangulation. The main difference is that as far as predicates are concerned the projections are not on the guide, but on the side of their source. The CDT on n points and m constraints, including projections, can be constructed in $O(N \log N)$ time, where $N = n+m$.

Some applications may require the shape of the same sites and guides for various α values. For example, we may want to know the smallest α for which the area of the guided α-shape is at least x. Unfortunately, as shown in Subsection 3.2, the guided α-shape at a certain α-value contains a subset of the projections; adding the part of the projections to an existing CDT is not known to be more efficient than recomputing the complete CDT.

If the value of interest is a monotone function in the value of α, such as the area of the shape, there is an easy way to compute it: create a sorted list of α values at which the guided α-shape changes and perform a binary search on this list. In this case, the appropriate CDT must be recomputed for each of the $O(\log N)$ α-values considered. This leads to a total running time of $O(N \log^2 N)$. Inspired by Guibas *et al.* [GKS92] and Chazelle *et al.* [CDH*02], we can lower this bound to $O(n \log N + m \log^2 N)$. This may not seem like a major improvement, but note that in practical situations the number of constraints m is much smaller than the number of sites n.

3.5. Artifacts arising in practice

There are a few artifacts that occur when we apply the guided α-shape method to bound the surfaces for urban reconstruction. Notably, noise and outliers can cause measurement points on the other side of the boundary. The method can be used without change to reconstruct multiple shapes per plane, as long as the point sets are at least 2α apart.

Factors like noise may cause points outside a surface boundary, beyond a guide. Any method for geometric reconstruction should be able to handle these artifacts. While our method is able to handle these points without breaking down, it can also identify the noise and reduce its impact on the result. Because we incorporate guides to indicate preferable locations for boundary edges, we have an estimator for which regions are noisy. The guides partition the shape into components. Components that are very thin compared to the point distribution, are likely caused by noise (see Figure 11, region 5). These components are easily identified by comparing their area to the length of their guides.

Another common source of artifacts in geometric reconstruction is outliers: points that were incorrectly assigned to a cluster. In the case of urban reconstruction, outliers are mainly caused by vegetation, but points measured through windows are another source. The parts of the shape with outliers are usually significantly less dense and in most cases a neighboring surface provides a convenient guide to separate out the outliers. Our method will automatically incorporate the guide into the boundary, thereby separating the real surface from the outliers. This separation makes identifying regions with lower density easy.

4. Results

The method presented in Section 3 is applied to synthetic and real dense LiDAR data.

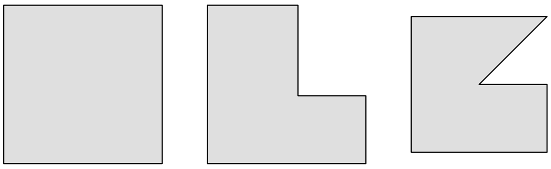


Figure 9: *The different shapes used in the tests on synthetic data; from left to right: square, L-shape, C-shape.*

4.1. Synthetic data

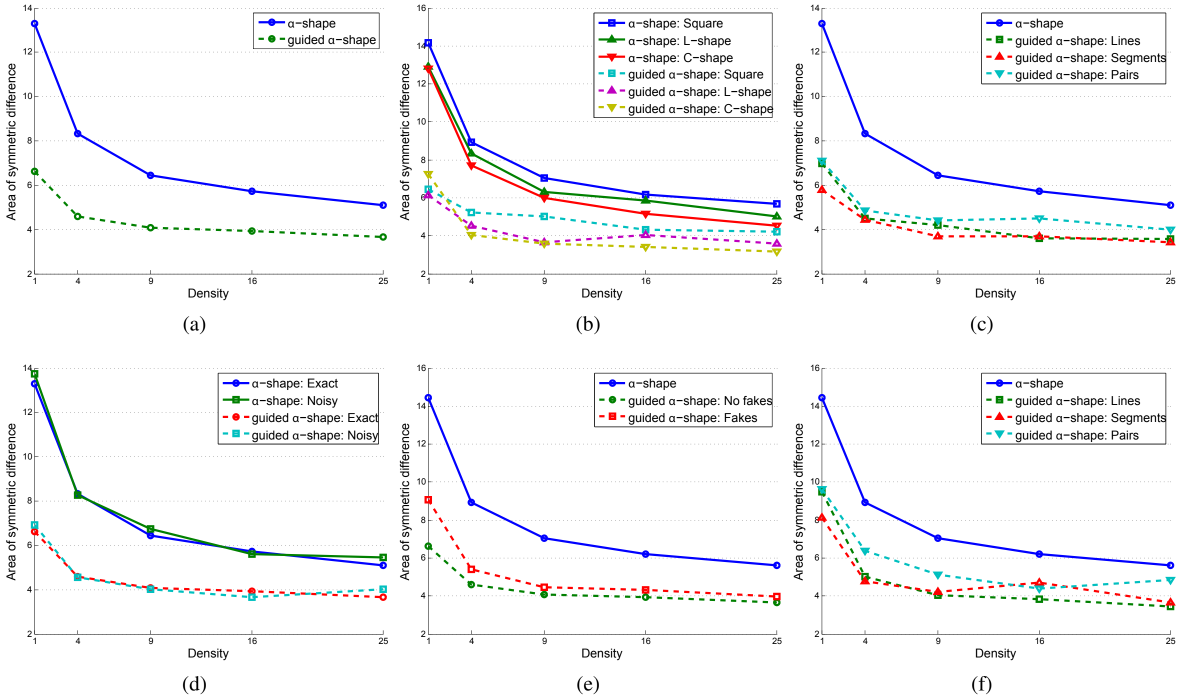
Our goal of running experiments on synthetic data is to ascertain whether the guided α-shape is better than the α-shape at reconstructing a shape from a point sample of its interior. The quality of either method is measured by the area of the symmetric difference between the produced shape and the original shape. We approach our goal by answering five questions on the quality of the shape produced by the different methods.

1. Is the guided α-shape better than the α-shape when guides are supplied?
2. Does the type of shape influence the individual quality of the α-shape, guided α-shape, or their relative quality?
3. Does the type of guides influence the quality of the guided α-shape?
4. Does the addition of a Gaussian noise model influence the individual quality of the α-shape, guided α-shape, or their relative quality?
5. Does the inclusion of a number of additional false guides influence the quality of the guided α-shape?

Each of these questions is visualized in a graph and tested using a t-test with $(p < 0.05)$ as criterion for rejecting the null hypothesis that both results were sampled from the same distribution. In the rest of this subsection, by ‘significant’ we indicate that the t-test rejects the null hypothesis.

All tests are done by comparing the α-shape to the guided α-shape on different sets of points and guides. These sets are created by sampling the interior and edges of a polygon. The interior is uniformly sampled up to a certain point density and each edge has 50% chance of being included as a guide. To incorporate variation in the type of shape, we use three different polygons to sample from, shown in Figure 9.

Because both the α-shape and guided α-shape are defined locally, we limit the number of shapes used. However, to cover a variety of cases we have included both shapes of varying simplicity and with different corner types (i.e. right convex, right concave, sharp convex and sharp concave). We expect any error in the α-shape and guided α-shape to be near the boundary of the shape. Therefore, we have normalized the errors by giving all shapes the same boundary length of 28. This leads to the interior areas of 49, 37, and 30.95 for the square, L-shape, and C-shape respectively.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 10: *The area of symmetric difference between the original shape and either the* α*-shape or the guided* α*-shape.*

The type of guides used may have an important influence on the guided α-shape. We base the guides on the selected edges of the shape in three different ways. We either supply the supporting line of the edge as guide (Lines), or we give the edge itself as guide (Segments), or we simulate a guide with a gap in it by cutting the edge into three segments of equal length and using the outer two as guides (Pairs).

For most tests, we directly use the point samples of the interior of the original shape at different densities. However, in practical applications, noise is an unavoidable problem. Therefore, we have created samples with added Gaussian noise. The coordinates of each sample point are perturbed by adding a random value from a normal distribution with $\mu = 0$ and $\sigma = 0.05$. This noise conforms with the typical noise in real data. This noise may have influence mainly by extending the α- and guided α-shapes over the boundary of the original shape.

Finally, we expect that guides have a large influence on the quality of the guided α-shape. While correct guides should force the shape to be more like the original, incorrect guides may create holes, or guide the shape too far outside the shape. This simulates real data, where we are not sure of the correctness of the guides.

To measure the impact of incorrect guides, we have added five fake guides to each sample. Note that this means that most input sets have more fake guides than the expected number of correct guides (i.e. half of the original edges). Each fake guide is created by taking a random point p inside the shape and translating it in a random direction by a third of the shape diameter. The guide contains p and has an independent random direction. If the guide has endpoints (Segments or Pairs setting), one endpoint is p and the other lies the average length of the real edges away from p. Intersecting guides are split into multiple guides.

Because the shape of a point set is dependent on the density of the sampling, each of the tests is done for five different sampling densities. The α-value of both the α-shape and guided α-shape is set to the average distance to the sixth nearest neighbor. This choice is based on the kissing number of unit disks in the plane, and we expect this α will result in shapes without holes.

The results for the different questions are shown in Figure 10. Most of these results are the average over 60 sample sets, but the overall results average 180 samples, and the fake guides per type average 20 samples. Whenever we indicate a (significant) difference, this is supported by t-tests.

Figure 10(a) shows the overall results. The guided α-shape consistently performs significantly better than the α-shape. The shape of the polygon does not have a significant impact on these results, as shown in Figure 10(b). The guided α-shape is better than the α-shape on each shape, and the only time the results within one method differ is for the α-shapes of the square and C-shape at density 9.

When we compare the influence of the different types of guides, as shown in Figure 10(c), t-tests support two observations. Firstly, the guided α-shape consistently performs significantly better than the α-shape. Secondly, the significance of the difference between the different ways of creating the guides depends on the density. However, for the densities of 9 and upward, the pairs perform the worst, and the results of the segments and lines are comparable.

When we introduce noise in the point coordinates or fake guides, paired t-tests indicate that the guided α-shape still always performs significantly better than the α-shape. A notable result is that within one method, noise in the points can actually improve the results, as is the case for density 25, shown in Figure 10(d).

Contrary to expectations, introducing fake guides does not make the guided α-shape perform worse than the α-shape, as shown in Figures 10(e) and 10(f). In most cases, the difference between the guided α shape with and without fake edges is significant. However, these shapes with fake edges still estimate the interior of the original polygon significantly better than the α-shape. For example, in Figure 10(f), the difference between the guided α-shape with the pairs setting and the α-shape at density 25 seems small, but the paired t-test gives a p-value of $5.79 * 10^{-7}$.

4.2. Real data

We have compared the α-shape and guided α-shape on airborne LiDAR data, which is frequently used for urban reconstruction. First, we applied efficient RANSAC [SWK07] to partition the point set into planar clusters. Each cluster is projected onto its supporting plane to provide the input for both methods. The guided α-shape also receives the intersection lines between planes that have a point within 1 meter of both clusters. The α is the same for both methods and set to 0.5, based on the data distribution.

The boundary of the surfaces cannot be directly inferred from the planes and their intersections. Even with this knowledge, determining the intersection parts to use is not straightforward. Besides identifying the parts of the intersections to use for the boundary of the surface, our algorithm also generates a fitting boundary in regions without appropriate intersection lines.

Constructing the shapes from a raw LiDAR data set of two million points took roughly 32 minutes on a consumer computer. This resulted in a model containing 101 planar shapes. Roughly half the points were classified as outliers, i.e. vegetation or surfaces too small to confidently reconstruct. As part of this process, computing the guided α-shapes of all surfaces in the data took seven minutes. Computing the α-shapes of the same surfaces requires two and a half minutes.

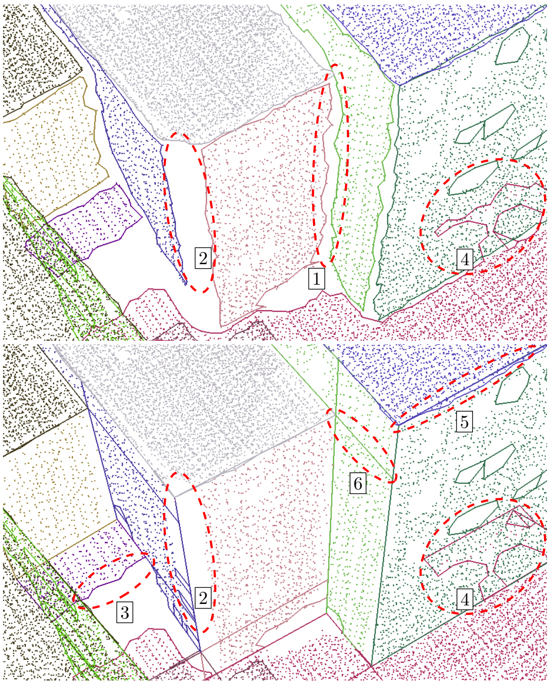


Figure 11: *The α-shapes (top) and guided α-shapes (bottom) of part of a LiDAR data set, with interesting regions marked.*

Figure 11 shows both the α-shape and guided α-shape of a number of surfaces. The shapes differ in some key aspects. The α-shape has jagged ‘cracks’ between neighboring surfaces (1). In the guided α-shape these neighboring surfaces share long boundary edges. Some cracks (2) may indicate a missing surface, but the lack of data may just as well be caused by invisibility of part of the surface. Because both surfaces are close enough to their intersection line, the guided α-shape uses this line, but only if no other surface is closer to the point set. In regions without intersection lines, the guided α-shape is the same as the α-shape (3).

Besides constructing boundaries that are more appropriate for urban reconstruction, the guided α-shape can also be used to identify artifacts in the data (4: red lines) and (5). The α-shape will not recognize these artifacts, making identifying them a difficult problem. The guided α-shape automatically separates these unwanted points from the main surface shape. This makes identifying and removing them straightforward: points measured through windows (4) usually give a sparser sample, and noise near a surface boundary (5) produces thin shapes near a guide.

A type of artifact specific to the guided α-shape is that the intersection lines may divide one surface into multiple shapes (6). However, after identifying unwanted shapes (eg. 4 and 5), it is easy to remove these edges (6) by joining the remaining shapes of a surface back together.

5. Conclusions

We have presented the guided α-shape, the shape of a set of n points and m line segments, and an algorithm to construct it in $O((n+m)\log(n+m))$ time. A comparison between the guided α-shape and the α-shape reveals that the guided α-shape is a better estimator for the interior of a point sampled polygon. This remains true for different shapes and when incorrect guides are provided. We have shown that it is not necessary to provide an accurate estimate of the appropriate boundary vertices; their supporting lines are sufficient.

A few interesting problems remain to be addressed. The guided α-shape has one parameter. Like in the α-shape, the choice of this parameter has an important influence on the quality of the final shape. It remains an open problem what the best way is to determine the value of α [CGPZ05,TC98]. This problem also affects the guided α-shape. However, we expect that as long as α is not prohibitively small, the guides have a much larger influence than α.

We have applied the guided α-shape to urban reconstruction. For this application, it is safe to assume that surfaces are planar and guides are straight. However, the guided α-shape may be appropriate for computing the shape of a point set in other situations as well. These situations may use curved guides or curved surfaces. Extending the guided α-shape to encompass these shapes is an interesting problem.

Other applications may pose different challenges. Indoor scenes and engineered objects are usually composed of simple surfaces. Reconstructing them as such may prove beneficial. These scenes may include parts that cannot be correctly reconstructed using only simple shapes. In this case, the non-guide parts of the guided α-shape may give good edges to connect a mesh that reconstructs the irregular parts.

Finally, in our real world data sets, noise and outliers are a recurring problem. This may cause input points to lie outside the desired shape. An automatic way of detecting these parts of the shape should improve the reconstruction results and the guided α-shape may prove helpful for resolving this.

Acknowledgments This research has been supported by the GATE project, funded by the Netherlands Organization for Scientific Research (NWO) and the Netherlands ICT Research and Innovation Authority (ICT Regie).

References

[ABE98] Amenta N., Bern M., Eppstein D.: The crust and the β-skeleton: combinatorial curve reconstruction. *Graphical Models and Image Processing 60*, 2 (1998), 125–135. 2

[ACSTD07] Alliez P., Cohen-Steiner D., Tong Y., Desbrun M.: Voronoi-based variational reconstruction of unoriented point sets. In *SGP* (2007), pp. 39–48. 1

[CDH*02] Chazelle, Devillers, Hurtado, Mora, Sacristan, Teillaud: Splitting a Delaunay triangulation in linear time. *Algorithmica 34* (2002), 39–46. 7

[CGPZ05] Cazals F., Giesen J., Pauly M., Zomorodian A.: Conformal alpha shapes. In *PBG* (2005), pp. 55– 61. 2, 10

[Che89] Chew L. P.: Constrained delaunay triangulations. *Algorithmica 4*, 1 (1989), 97–108. 2, 6

[EKS83] Edelsbrunner H., Kirkpatrick D. G., Seidel R.: On the shape of a set of points in the plane. In *IEEE Transactions on Information Theory* (1983), vol. 29, pp. 551–559. 2

[ET92] Edelsbrunner H., Tan T. S.: An upper bound for conforming Delaunay triangulations. In *SoCG* (1992), pp. 53–62. 3

[GKS92] Guibas L., Knuth D., Sharir M.: Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica 7* (1992), 381–413. 7

[KBSS01] Kobbelt L. P., Botsch M., Schwanecke U., Seidel H.-P.: Feature sensitive surface extraction from volume data. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2001), ACM, pp. 57–66. 2

[Kir79] Kirkpatrick D. G.: Efficient computation of continuous skeletons. In *FoCS* (1979), pp. 18–27. 2

[LWY08] Ling R., Wang W., Yan D.: Fitting sharp features with loop subdivision surfaces. In *SGP* (2008), pp. 1383–1391. 2

[Rot03] Rottensteiner F.: Automatic generation of high-quality building models from lidar data. *IEEE Computer Graphics and Applications 23*, 6 (2003), 42–50. 2

[Rup95] Ruppert J.: A delaunay refinement algorithm for quality 2-dimensional mesh generation. *Journal of Algorithms 18*, 3 (1995), 548–585. 3

[SDK09] Schnabel R., Degener P., Klein R.: Completion and reconstruction with primitive shapes. *Computer Graphics Forum 28* (2009), 503–512. 2

[SWK07] Schnabel R., Wahl R., Klein R.: Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum 26*, 2 (2007), 214–226. 2, 9

[SYM10] Salman N., Yvinec M., Merigot Q.: Feature preserving mesh generation from 3d point clouds. *Computer Graphics Forum 29*, 5 (2010), 1623–1632. 2

[TC98] Teichmann M., Capps M.: Surface reconstruction with anisotropic density-scaled alpha shapes. In *VIS* (1998), pp. 67–72. 2, 10

[TTC07] Tseng Y.-H., Tang K.-P., Chou F.-C.: Surface reconstruction from LiDAR data with extended snake theory. In *EMMCVPR '07* (2007), pp. 479–492. 1, 2

[Vel92] Veltkamp R. C.: The γ-neighborhood graph. *Computational Geometry Theory and Applications 1*, 4 (1992), 227–246. 2

[VGSR04] Vosselman G., Gorte B. G. H., Sithole G., Rabban T.: Recognising structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 46* (2004), 33–38. 2

[vKvLV11] van Kreveld M., van Lankveld T., Veltkamp R. C.: *On the shape of a set of points and lines in the plane*. Tech. rep., Utrecht University, Department of Information and Computing Sciences, 2011. 4

[YHNF03] You S., Hu J., Neumann U., Fox P.: Urban site modeling from LiDAR. In *ICCSA* (2003), vol. 2669/2003 of *LNCS*, pp. 579–588. 2

[ZN08] Zhou Q.-Y., Neumann U.: Fast and extensible building modeling from airborne LiDAR data. In *SIGSPATIAL* (2008), pp. 1–8. 2