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

Marc van Kreveld

Thijs van Lankveld

Remco Veltkamp

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

Marc van Kreveld, Thijs van Lankveld, and Remco C. Veltkamp

June 10, 2011

Abstract

Detailed geometric models of the real world are in increasing demand. LiDAR data is appropriate to reconstruct such models. For urban reconstruction, it works well to reconstruct the individual surfaces of the scene and connect them 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 generalized and guided α-shape: two extensions of the well known α-shape. The generalized α-shape handles lines (generalized points) as if they are infinite collections of points. The guided α-shape handles lines (guides) as indicators for 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, for an input of n points and m guides, both the generalized and guided α-shape can be constructed in $O((n + m) \log(n + m))$ time. 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.

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 two novel structures, the generalized α-shape and the guided α-shape, that construct the boundary of a set of planar points and lines. Both structures are similar to the α-shape, but the generalized α-shape regards lines as infinite point sets, and the guided α-shape regards lines as indications for appropriate locations for the shape boundary, see Figure 8.

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 generalized α-shape. While this shape is interesting in theoretical sense, it is not very suited for surface reconstruction. The guided α-shape, defined in Section 4, is more appropriate for surface reconstruction. We prove that the guided α-shape can be constructed as efficiently as the α-shape in Subsection 4.4. We compare the quality of the resulting shape to the α-shape in Section 5. Finally, we conclude the paper and present directions for further research in Section 6.

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 [1, 19, 23]. 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
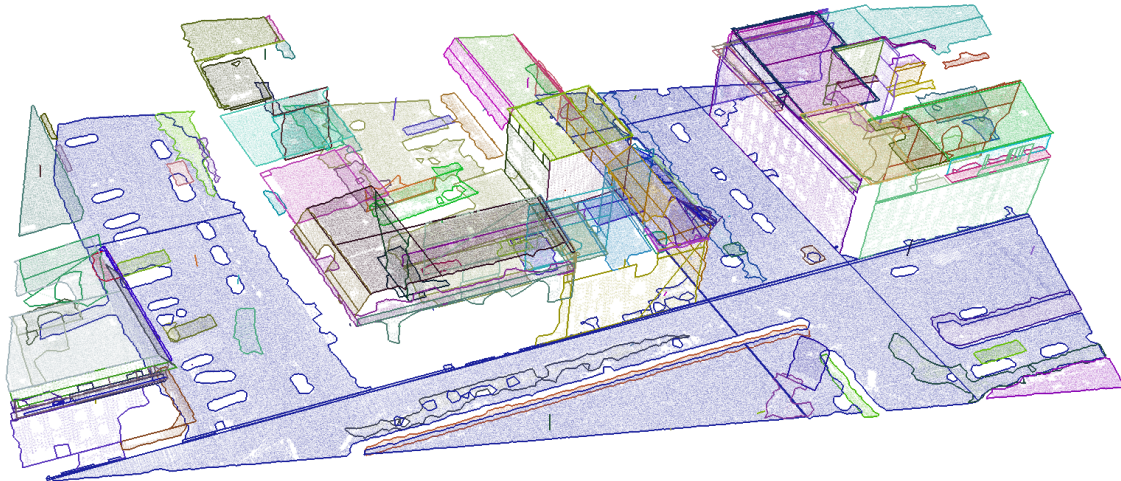
Figure 1: The guided α-shapes in an urban LiDAR scene. These connect neighboring surfaces along an edge.

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 [9, 14, 18]. 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 Li-DAR data [10, 16, 25, 26, 27]. These methods exploit the abundance of planar surfaces, but they usually assume a regular grid in the horizontal plane containing one height measurement per cell. This leads to two problems. Firstly, these methods need complicated procedures to compensate for irregular sampling causing empty grid cells. Secondly, they are not able 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 [21] and region growing [23]. Schnabel *et al.* [20] have used these surfaces to reconstruct the object as a manifold. They use a version of marching cubes [12] to reconstruct the manifold. This has the disadvantage of greatly reducing the simplicity of the planar surfaces. By comparison, we wish to retain the conciseness of planar surfaces. This leaves the problem of reconstructing the boundary of each surface. This is solved by identifying the shape of the subset of points that were measured in each surface. In many cases this shape reconstruction process can be guided by the intersection lines of the surface with its neighbors, because after reconstruction these should fit together neatly to form a watertight geometric model of the scene.

Even though related work does not clearly define what the shape of a point set actually is, different methods have been proposed to find a region containing a point set. Structures like the crust, β-skeleton and γ-neighborhood graph [2, 24] assume the points were sampled from one closed curve, 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 [3, 6, 15]. More complex methods can produce the shape of non-evenly distributed point sets [15, 22].

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

Figure 2: The shape of a tree at two different scale levels is shown by the blue outline. At a low scale level (left) the shape of the tree is very rough, while at a higher scale level (right) the shape contains much more detail.

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 [5] and generalized Voronoi diagrams [11]. 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.

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. Figure 2 shows an example of the influence of a scale parameter.

Edelsbrunner *et al.* [6] 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 3. 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 [6].

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
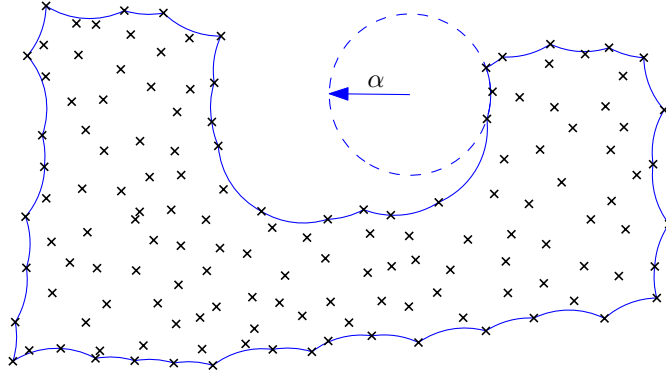
3

Figure 3: The α-hull. Its complement contains all empty α-disks. The α-shape is the straight line version of the hull.

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 $\leq \alpha$.

The α-shape can be influenced by lines in three basic ways. When we say 'lines', the concept can generally also be applied to half-lines and line segments.

Firstly, the lines may denote infinite clusters of sample points. In this case we call the lines *generalized points* and the lines should be interior to the *generalized α-shape*. This case is similar to the way line segments are handled in the Voronoi diagram generalized to points and line segments [11]. We describe a method for determining the α-shape of a set of generalized points in Section 3. This simple adaptation of the α-shape will always result in a shape containing the line segments. However, we will show that some artifacts that occur mainly in corners, make this approach less than optimal when we are interested in computing a shape using the line segments as boundary.

Secondly, the line segments may indicate likely locations of the boundary of the shape. In this case we call the segments *guides* and the boundary of the *guided α-shape* should follow the line segments where appropriate, as described in Section 4. This problem is the main subject of this paper, because the method will result in a polygon suited to bounding a surface in the geometric reconstruction process. These results differ from the regular α-shape on another point: the vertices of the shape are not necessarily part of the point set. Experimental results on the guided α-shape are presented in Section 5.

Thirdly, the line segments may indicate forbidden regions where the shape is not allowed to be, without directly indicating the boundary of the shape. In this case we call the segments *shields* and the line segment must be outside the shape. This shape may be interesting for applications like geometric interpolation of earth surface samples in the presence of fault lines. Because computing this shape has proven to be very complex and because the results are less interesting for our application, this problem is left as future research. The reason for addressing it here is just for completeness.
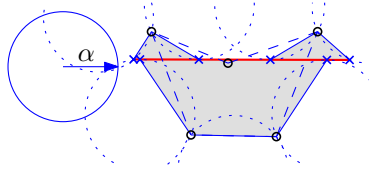
4

Figure 4: The generalized α-shape of a point set does not always cover a larger region than the shape without line segments. The black circles are the points and the thick red line segment is added in the generalized α-shape. The dashed blue lines are in the α-shape and the generalized α-shape consists of the solid blue lines.

3 Generalized α-shape

Lines can indicate the interior of the shape. In this case the shape of a collection of points and lines contains all points and lines in its interior. Inversely, the exterior of the shape is directly related to the union of all empty disks of a fixed radius. This structure is in many ways similar to the α-shape.

Definition 2 (Generalized α-shape)**.** Given a point set S and a set of lines L, the generalized α-shape of $S \cup L$ is the α-shape of $S \cup \{x | x \in l \in L\}$.

Similar to Observation 1, the generalized α-shape separates the internal and external faces of the shape.

Lemma 1. *The generalized α-shape a set of point S and a set of lines L partitions the plane into internal and external faces.*

Proof. The generalized α-shape is equal to the α-shape of $S \cup \{x | x \in l \in L\}$. The remainder of the proof follows straightforward from Observation 1 by extending the notion of the Delaunay triangulation to infinite point sets. □

It seems safe to assume that the generalized α-shape covers a superspace of the α-shape without line segments. However, just like adding points to the α-shape can reduce the region it covers, adding line segments to the shape can result in smaller interior faces, as shown in Figure 4.

For generalized α-shapes, the main problem is modeling the effects of the infinite collection of points on the lines without reducing the computational efficiency of constructing the shape.

3.1 Constructing the generalized α-shape

We will construct the generalized α-shape using line segments instead of lines, because we make use of the medial axis and its definition is not extended to lines. However, the extension to lines is straightforward. Just replace lines by line-segments with endpoints located very far away.

Conceptually, the generalized α-shape considers the line segments as an infinite amount of points to add to the point set. In practice, the boundary of the generalized α-shape is composed of a number of edges linear in the number of points n and line segments m, and can be constructed in $O((n + m) \log(n + m))$ time, using $O(n + m)$ space. The process is shown in Figure 5. An important observation on increasing the efficiency is that there is a limited number of points on the line segments that influence the shape: all empty open disks with a point from a line segment on its boundary must either be tangent to the segment or the disk only touches one of the endpoints of the segment.

Theorem 1. *The generalized α-shape of n points S and m line segments L can be constructed using $O((n + m) \log(n + m))$ time and $O(n + m)$ space.*
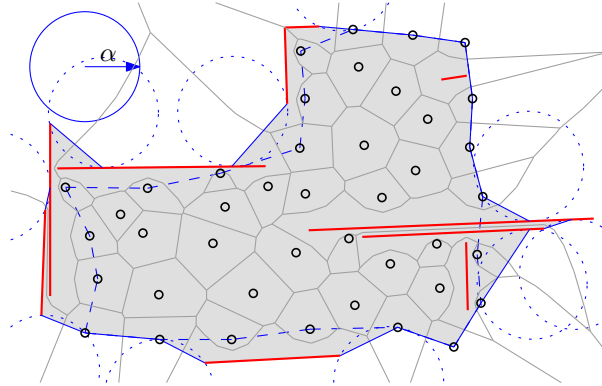
Figure 5: The construction of the generalized α-shape. The black circles are the points and the thick red line segments are the line segments. The dashed blue lines are in the original α-shape and the line segments add the solid blue lines between the blue crosses that are at distance α from the medial axis.

Proof. The generalized α-shape has an edge between each pair of points on the boundary of an empty disk with radius α. These edges can be divided into three categories: point-point, line-point, and line-line. The case of point-point edges is the same as an edge formed in the regular α-shape, with the addition that the disk cannot intersect a line segment in its interior.

In the case of an edge connecting the infinite point set represented by a line segment to a point $p \in S$, the edge will be incident to either the endpoint of the segment or a point where a disk with p on its boundary is tangent to the line. Otherwise the disk must intersect the segment in its interior meaning it is not empty.

In the case of an edge connecting two line segments, both vertices can be either an endpoint of a segment or a point where the empty disk is tangent to the line.

In all three cases the center of the disk has distance α to both incident vertices, whether point or line segment. This information is captured in the medial axis of $S \cup L$.

The medial axis of N points and line segments can be computed in $O(N \log N)$ time [13] and contains $O(N)$ straight or parabolic edges. A straight or parabolic edge can have at most two points at distance α to the elements it separates. A brute force search of all edges in the medial axis will produce the generalized α-shape in $O(n + m)$ time from the medial axis. $\qquad \square$

4 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, so we have 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.

An α-projection is the point on a guide closest to a site, its source, if it conforms to some conditions, as shown in Figure 6. These conditions, listed in Definition 4, are chosen to elegantly identify appropriate parts of the guides for the shape boundary. Where the point set is near a guide, the α-projections are the vertices of the shape boundary. This is achieved by using the
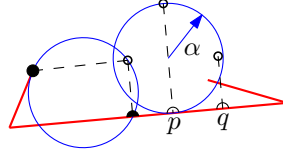
Figure 6: The black (half) disks are α-projections. The black semi-circles are not α-projections, because one of the conditions of Definition 4 is not met: p does not share an empty connected disk with its source and the segment connecting q to its source intersects a guide.
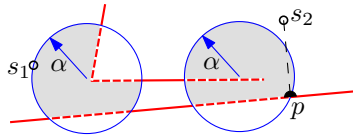

Figure 7: Connected α-disks of site s_1 and a projection p of s_2 are indicated by the gray regions. The dashed parts of the guides are part of the connected α-disks, but the circular blue boundaries are not.

α-projections as additional points, similar to the Steiner points added in conforming Delaunay triangulations [7, 17]. However, unlike Steiner points, all projections lie infinitesimally close to a guide and strictly to one side of it.

Definition 3 (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 7).

Definition 4 (α-projections). Given a set of sites S, a set of non-intersecting guides G, and a positive real value α, $P_{S,G}^{\alpha}$ is a set of points near G, called α-projections. Each α-projection $p \in P_{S,G}^{\alpha}$ 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 $x \leq \alpha$.

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 4. 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.
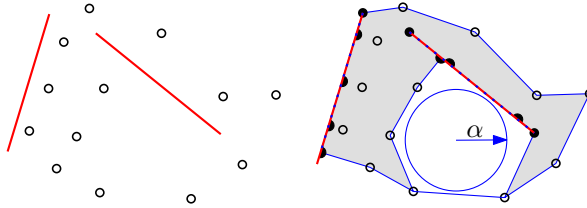
Figure 8: A set of sites and guides and the line-guided α-shape of this set. The (half) disks are α-projections.

Armed with the definitions for the connected α-disk and projections, we can now define the line-guided α-shape, shown in Figure 8. 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 5 (Line-guided α-shape). Given a set of sites S, a set of non-intersecting guides G, and a positive real value α, let $P_{S,G}^{\alpha}$ denote the set of α-projections of S onto G.

A point $s \in S \cup P_{S,G}^{\alpha}$ is α-extreme if there exists a connected α-disk of s empty of sites and α-projections. Two points in $S \cup P_{S,G}^{\alpha}$ 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 4.1. 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 4.2. In Subsections 4.2 to 4.5, we present our algorithm to construct the guided α-shape and we discuss some artifacts arising in practice.

4.1   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 9.

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
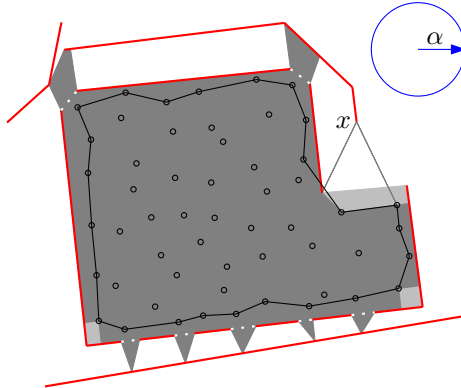
Figure 9: An example of realistic input where the line-guided α-shape (dark gray) gives undesired results. Slightly adjusting the criteria near guide endpoints gives the guided α-shape. This shape does not cross the small gaps indicated by the dotted lines, it covers the light gray regions, and it does not connect to x. The original α-shape is black.

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.

Definition 6 (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 regions or faces from exterior faces. The interior faces do not contain the center of a valid connected α-disk. Similar to the α-shape, the interior faces of the guided α-shape are the union of the triangles in the triangulation of the sites, guides, and projections that have a circumdisk with radius smaller than α. This can be proven comparable to how Edelsbrunner *et al.* [6] prove the faces of the α-shape are a subset of the Delaunay triangulation.

Lemma 2. *The interior faces of the guided α-shape are the union of the triangles of the constrained Delaunay triangulation C on the sites, guided, and projections with a circumdisk of radius $x < \alpha$.*

Proof. It is straightforward to see from the definition of the α-neighbors that the edges and vertices of the guided α-shape are a subset of the edges and vertices of C. The edges of the guided α-shape separate interior from exterior faces. It is not uncommon for some of these exterior faces to be degenerate. For example, because projections are infinitesimally close to a guide, there are degenerate (area 0) faces between the projections and guides.

An interior face can be identified from the fact that it does not contain the center of a valid connected α-disk. The triangles in C that do not contain the center of an valid connected α-disk must have a circumdisk with radius smaller than α. By contrast, an exterior face only contains triangles of the CDT with a circumdisk with radius at least α: the triangles that contain the center of a valid connected α-disk. The edge between an interior and exterior triangle in C must connect two α-neighbors by definition 6. □
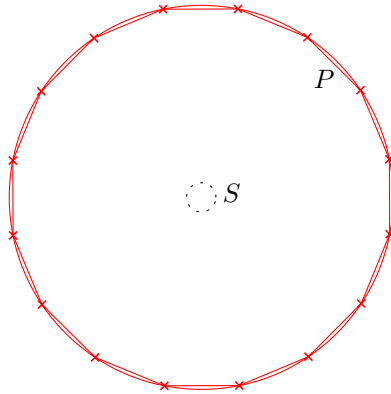
Figure 10: An example resulting in $\Theta(n * m)$ projections if validity is not enforced. The n sites are evenly spread over dotted circle S and all endpoints of the m guides are spread evenly over the red circle P. If validity is not enforced, each guide contains projections for half of the sites.
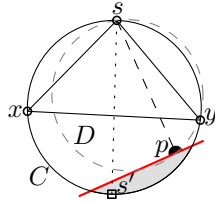

Figure 11: The different structures used in the proofs of Lemmas 3 and 4. Site s is a source of projection p, x and y are points adjacent to s and sharing the CDT circumcircle C shown in black. The dashed gray circle D shows p is actually a projection. The point opposite of s on the disk is s'. The gray region is the part of the disk behind the guide.

4.2   Number of projections

Validity, as described in Definition 6, 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, as shown in Figure 10. 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 11 shows most of the structures used in the proofs.

Lemma 3. *In the CDT of the sites and guides, each projection is inside the circumdisk of a triangle incident to its source.*

Proof. For any projection p and any of its sources s, take a disk D with radius $x \leq \alpha$ that has s and p on its boundary and that has a connected component C that does not contain any sites or guide endpoints, and has s on its boundary. Note that according to Definition 4, such a disk must exist for each pair of projection and source and according to Definition 3 C is an connected x-disk of s touching p.
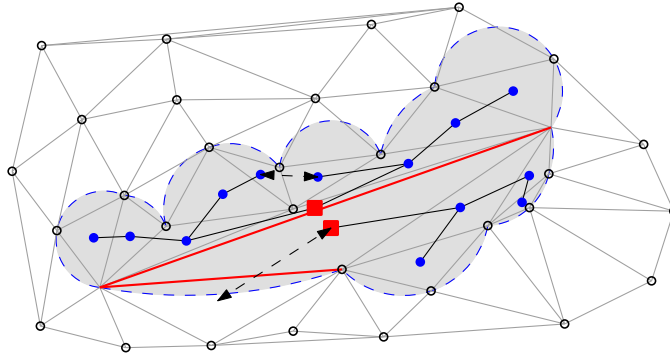
10

Figure 12: 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).

Let sxy be the unique triangle of the CDT intersected by the line segment ps. If p lies on xy, it is trivially inside the circumdisk of sxy. Otherwise, sp intersects xy and xy cannot be a constraint. This means that for any circumdisk Q of sp, if x or y is inside Q, then it must be in the same component of the connected disk in Q as s and p. Because C is a valid connected disk, D cannot contain x or y. Therefore, the circumdisk of sxy must contain p. □

Lemma 4. *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. □

4.3 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 4.4. We call this structure a *cumulus tree*, based on its cloud-like appearance. The two cumulus trees of a constraint are shown in Figure 12

Definition 7 (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 5 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 5. *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 $m \leq n$. $\square$

Lemma 6. *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 $O(n + m)$.*

Proof. This proof follows straightforward from the proof that there are $O(n+m)$ pairs of triangle and constraint for which the circumdisk of the triangle overlaps the constraint and the constraint is visible to the triangle.

If a CDT triangle's circumdisk overlaps a constraint, there is a disk D completely inside the circumdisk that is tangent to the constraint and touching a vertex of the triangle. This means that the center of D is equally distant to the vertex and the constraint and so this vertex and constraint share an edge in the segment-generalized Voronoi diagram [11] of the vertices and constraints. As this diagram contains $O(n + m)$ edges and each of these edges bounds two cells, there are $O(n + m)$ pairs of triangles and constraint for which the circumdisk of the triangle overlaps the constraint. $\square$

4.4   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 [5].

- 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.

Algorithm 1 Computing the projections of the guided α-shape.

PROJECT(e, t, c, P).

---

input e: an edge in the CDT, t: a triangle incident to e, c: a constraint, and P the current
collection of projections onto c.

$\quad s \leftarrow$ the vertex of t opposite to e
$\quad D \leftarrow$ the circumdisk of t
$\quad$ **if** $c = e$ **or** c intersects D on opposite side of e to s **then**
$\quad\quad p \leftarrow$ the point on c closest to s
$\quad\quad$ **if** p is inside D **and** p is not an endpoint **then**
$\quad\quad\quad s$ projects on D: add p to P
$\quad\quad$ **end if**
$\quad\quad e_{cw} \leftarrow$ the edge of t clockwise to s
$\quad\quad$ **if** e_{cw} is not a constraint **then**
$\quad\quad\quad t_n \leftarrow$ the triangle incident to e_{cw} that is not t
$\quad\quad\quad$ PROJECT(e_{cw}, t_n, c)
$\quad\quad$ **end if**
$\quad\quad e_{ccw} \leftarrow$ the edge of t counter clockwise to s
$\quad\quad$ **if** e_{ccw} is not a constraint **then**
$\quad\quad\quad t_n \leftarrow$ the triangle incident to e_{ccw} that is not t
$\quad\quad\quad$ PROJECT(e_{ccw}, t_n, c)
$\quad\quad$ **end if**
$\quad$ **end if**

- 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.

4.4.1 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 pseudo code for computing the projections is given in Algorithm 1. This recursive algorithm 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.

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 3 and 6 that this algorithm constructs all projections in $O(n+m)$ time.

4.4.2 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.
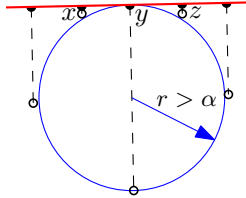
Figure 13: α-projections have a minimum α value; at smaller values they should not influence the guided α-shape.

Lemma 7. *All the $\tilde{\alpha}$ values at which a projection appears can be computed from the CDT in $O(n + m)$ time.*

Proof. For a projection p on an endpoint of a guide, $\tilde{\alpha}$ can be computed from the circumdisks of the triangles in the CDT p shares with its sources. Note that for any edge ps for which the centers of the circumdisks of its incident triangles are on opposite sides of ps, the circumdisk with ps as its diameter is also valid and smaller than either circumdisk.

For a projection p in the interior of a guide, it is less obvious which is the smallest connected disk D p shares with its source s. However, because of Lemma 3 we do know that there is a connected disk Q shared by p and s that is completely inside the circumdisk of a triangle t incident to s. Any connected disk shared by s and p that is smaller than Q must have its center closer to the line segment sp. Either the connected disk C with sp as diameter is valid, or there is a point x that also shares the boundary of D; x can be either a site or an endpoint of a guide. We use $D(y)$ to denote the circumdisk of the triangle associated with node y of the cumulus tree.

Because we know D is a connected disk shared by s and x and because D is valid, we also know that sx is an edge in the CDT. Because p is on constraint c and inside the circumdisk of t, we know that t has a node in cumulus tree $T(c)$. According to Lemma 5, we know that there is an extreme node y in $T(c)$, such that y is the node furthest away from the root of $T(c)$ that contains p in $D(y)$. We can now easily check if the disk R with diameter sp is valid. If the centers of Q and $D(y)$ lie on opposite sides of sp, then R must be valid, because $\{Q \cup D(y)\}$ is valid and $R \subseteq \{Q \cup d(y)\}$.

Once we have determined x, $\tilde{\alpha}$ is the radius of either C or the circumcircle of spx. Which of these two options it is, depends on whether C contains x. $\qquad \square$

The necessity of checking the size of the x-disk is shown in Figure 13. 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.

4.4.3 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 4.4.2, 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.* [8] and Chazelle *et al.* [4], 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.

Lemma 8. *Given n points and m guides, the optimal α for a function $f(\alpha)$ can be determined in $O(n \log N + m \log^2 N)$ time, where $N = n + m$, if $f(\alpha)$ is monotone in α and $f(\alpha)$ can be computed in linear time from the guided α-shape.*

Proof. Following the binary search paradigm, we can determine the optimal α by considering the guided α-shape for $O(\log N)$ values for α. To construct these guided α-shapes, we run steps 1-3 of our algorithm once, and repeat steps 4-6 for each alpha value. Steps 4 and 6 take $O(N)$ time per α value, but step 5 costs more time. According to Guibas *et al.* [8], inserting K new points into an existing Delaunay triangulation of size M will take update time $O(M \log K + K)$ when employing randomized incremental construction. In our case, $K = N$ and $M = m$. According to Chazelle *et al.* [4], given point sets Y, of size N, and $X \subseteq Y$ location time for randomized incremental construction of Delaunay triangulation $DT(X)$ in the presence of a known $DT(Y)$, is $O(N)$.

Both results are proven for Delaunay triangulations under randomized incremental construction. However, no algorithm is known for randomized incremental construction of constrained Delaunay triangulations. In case of our specific triangulations, these bounds still hold. This is because all points that are inserted are projections and all projections are next to a constraint. The update bound by Guibas *et al.* is based on flipping edges around a newly inserted point. As constraints cannot be flipped, there can only be less flips. The location bound by Chazelle *et al.* is based on deconstructing $DT(Y)$ in randomized vertex order and constructing $DT(X)$ in the reverse order. This bound does not directly hold for any CDT, because there is no clear and consistent way to randomly insert the constraints. However, in our CDT we can restrict the procedure to the triangles of the cumulus trees, which are locally unconstrained Delaunay. This puts the total running time at $O(\log N) * [O(M \log K + K) + O(N)] = O((m \log N + N) \log N) = O(n \log N + m \log^2 N)$. $\square$

4.5   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 is preferable if 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 and components that are very thin compared to the point distribution, are likely caused by noise (see Figure 17(b), region 5). These components are easily identified by computing the Hausdorff distance between the guides and the remaining boundary of the component.

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.
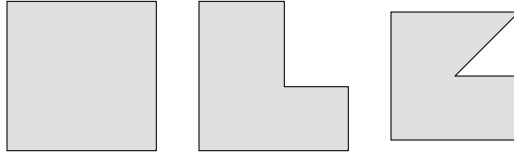
Figure 14: The different shapes used in the tests on synthetic data; from left to right: square, L-shape, C-shape.

5 Results

The method presented in Section 4 is applied to synthetic and real dense LiDAR data.

5.1 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 14.

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.

The type of guides used may have an important influence on the guided α-shape. We use three different ways of creating guides from the sampled edges of the original shape. The first is using the supporting line of the edge as guide (Lines), the second is using the edge itself as guide (Segments), and the third simulates 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
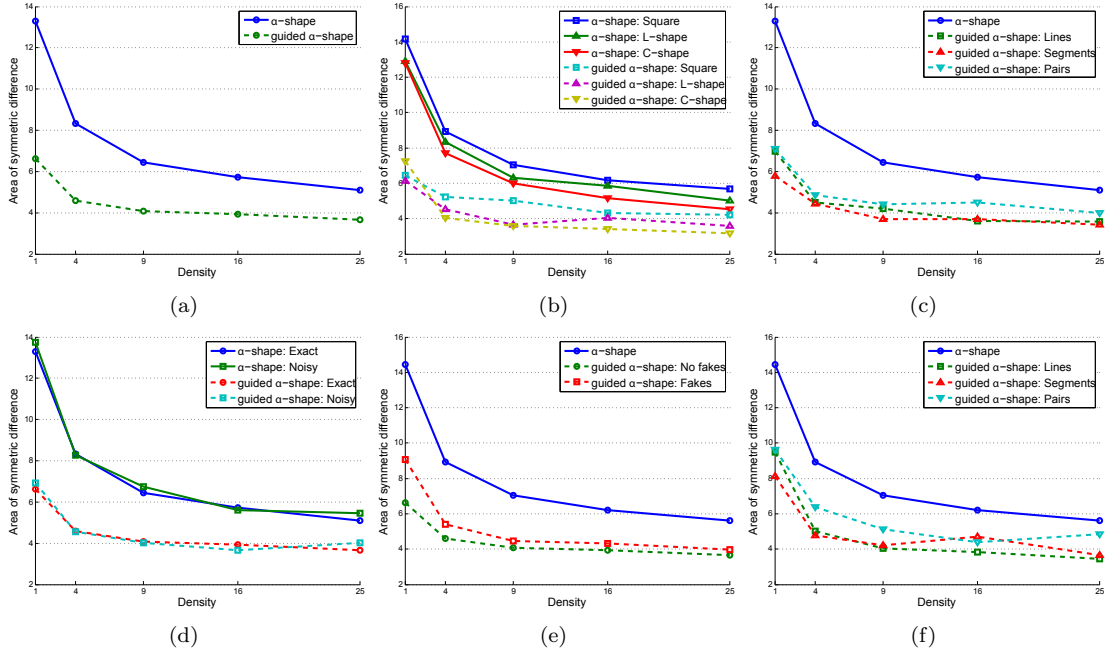
Figure 15: The area of symmetric difference between the original shape and either the α-shape or the guided α-shape.

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 15. 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. The α-shape and guided α-shape of one sample are shown in Figure 16.

Figure 15(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 15(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 15(c), t-tests support two observations. Firstly, the guided α-shape consistently performs significantly
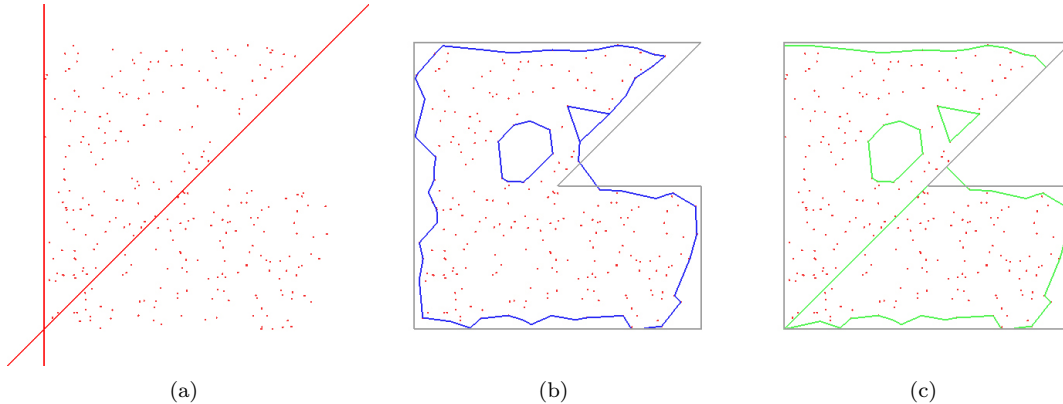
17

Figure 16: A set of points and lines (a) and the resulting α-shape (b) and guided α-shape (c) together with the sampled shape in gray.

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 15(d).

Contrary to expectations, introducing fake guides does not make the guided α-shape perform worse than the alpha-shape, as shown in Figures 15(e) and 15(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 15(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 $5.79 * 10^{-7}$.

5.2   Real data

We have applied our method to airborne LiDAR data, which is frequently used for urban reconstruction. We have applied efficient RANSAC [21] to partition the point set into planar clusters. We have calculated each surface's intersection lines with the other surfaces. Each cluster is projected onto its supporting plane to provide the input for the α-shape and guided α-shape. The guided α-shape also receives those intersection lines 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.

Note that the boundary of the surfaces cannot be directly inferred from the planes produced by RANSAC and their intersections. Even with this a priori knowledge, determining which parts of which intersections to use is not straightforward. Apart from 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 17 shows both the α-shape and guided α-shape of a number of surfaces. Some more comparisons are shown in Figure 18. 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
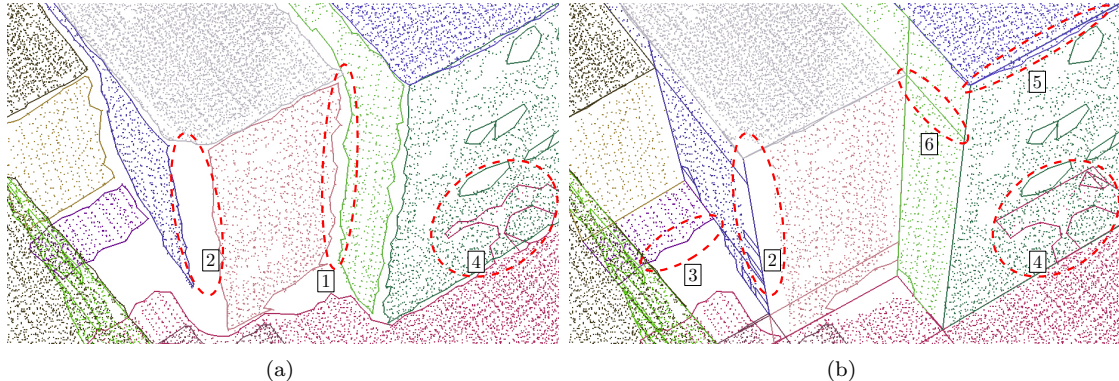
Figure 17: The α-shapes (a) and guided α-shapes (b) of part of a LiDAR data set, with interesting regions marked.

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 (6). 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.

6   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 [3, 15]. 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 the problem of 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 applications may include scenes that cannot be correctly reconstructed using only primitive shapes, like planes, cylinders, and spheres. 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.

References

[1] ALLIEZ, P., COHEN-STEINER, D., TONG, Y., AND DESBRUN, M. Voronoi-based variational reconstruction of unoriented point sets. In *SGP* (2007), pp. 39–48.

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

[3] CAZALS, F., GIESEN, J., PAULY, M., AND ZOMORODIAN, A. Conformal alpha shapes. In *PBG* (2005), pp. 55– 61.

[4] CHAZELLE, DEVILLERS, HURTADO, MORA, SACRISTAN, AND TEILLAUD. Splitting a Delaunay triangulation in linear time. *Algorithmica 34* (2002), 39–46.

[5] CHEW, L. P. Constrained delaunay triangulations. *Algorithmica 4*, 1 (1989), 97–108.

[6] EDELSBRUNNER, H., KIRKPATRICK, D. G., AND 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.

[7] EDELSBRUNNER, H., AND TAN, T. S. An upper bound for conforming Delaunay triangulations. In *SoCG* (1992), pp. 53–62.

[8] GUIBAS, L., KNUTH, D., AND SHARIR, M. Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica 7* (1992), 381–413.

[9] HOPPE, H., DEROSE, T., DUCHAMP, T., HALSTEAD, M., JIN, H., MCDONALD, J. A., SCHWEITZER, J., AND STUETZLE, W. Piecewise smooth surface reconstruction. In *SIGGRAPH* (1994), pp. 295–302.

[10] JIANG, X., AND BUNKE, H. Fast segmentation of range images into planar regions by scan line grouping. *Machine Vision and Applications 7*, 2 (1994), 115–122.

[11] KIRKPATRICK, D. G. Efficient computation of continuous skeletons. In *FoCS* (1979), pp. 18–27.

[12] KOBBELT, L. P., BOTSCH, M., SCHWANECKE, U., AND 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.

[13] LEE, D. T. Medial axis transformation of a planar shape. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Jul 1982), vol. 4, pp. 363–369.

[14] LING, R., WANG, W., AND YAN, D. Fitting sharp features with loop subdivision surfaces. In *SGP* (2008), pp. 1383–1391.

[15] MANDAL, D. P., AND MURTHY, C. A. Selection of alpha for alpha-hull in R^2. *Pattern Recognition 30*, 10 (1997), 1759–1767.

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

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

[18] SALMAN, N., YVINEC, M., AND MERIGOT, Q. Feature preserving mesh generation from 3d point clouds. *Computer Graphics Forum 29*, 5 (2010), 1623–1632.

[19] SCHICKLER, W., AND THORPE, A. Surface estimation based on lidar. In *ASPRS Annual Conference* (2001).

[20] SCHNABEL, R., DEGENER, P., AND KLEIN, R. Completion and reconstruction with primitive shapes. *Computer Graphics Forum 28* (2009), 503–512.

[21] SCHNABEL, R., WAHL, R., AND KLEIN, R. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum 26*, 2 (2007), 214–226.

[22] TEICHMANN, M., AND CAPPS, M. Surface reconstruction with anisotropic density-scaled alpha shapes. In *VIS* (1998), pp. 67–72.

[23] TSENG, Y.-H., TANG, K.-P., AND CHOU, F.-C. Surface reconstruction from LiDAR data with extended snake theory. In *EMMCVPR '07* (2007), pp. 479–492.

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

[25] VOSSELMAN, G., GORTE, B. G. H., SITHOLE, G., AND RABBAN, T. Recognising structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 46* (2004), 33–38.

[26] YOU, S., HU, J., NEUMANN, U., AND FOX, P. Urban site modeling from LiDAR. In *ICCSA* (2003), vol. 2669/2003 of *LNCS*, pp. 579–588.

[27] ZHOU, Q.-Y., AND NEUMANN, U. Fast and extensible building modeling from airborne LiDAR data. In *SIGSPATIAL* (2008), pp. 1–8.
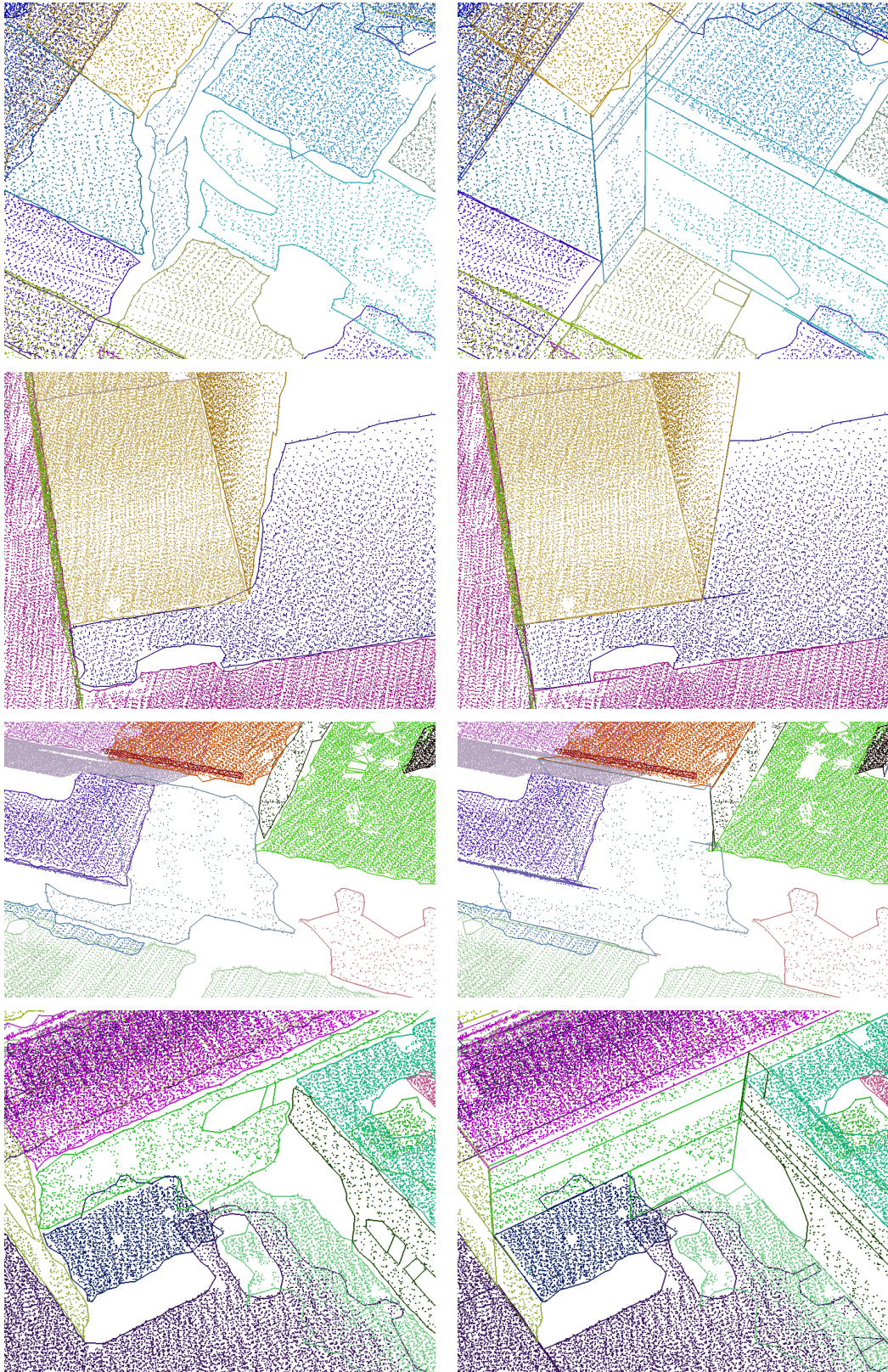
Figure 18: The α-shapes (left) and guided α-shapes (right) of parts of various scenes.