Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization

Alexandre Boulch, Martin de La Gorce, Renaud Marlet

Université Paris-Est, LIGM (UMR CNRS 8049), ENPC, F-77455 Marne-la-Vallée

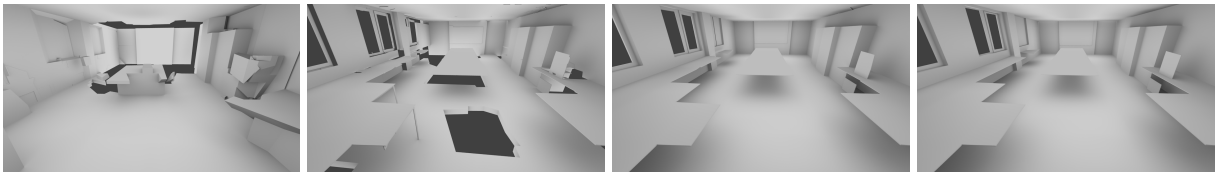


Figure 1: *Reconstruction by [CLP10], and by our method with regularization based on area, edge length and number of corners.*

Abstract

This paper presents a method for the 3D reconstruction of a piecewise-planar surface from range images, typically laser scans with millions of points. The reconstructed surface is a watertight polygonal mesh that conforms to observations at a given scale in the visible planar parts of the scene, and that is plausible in hidden parts. We formulate surface reconstruction as a discrete optimization problem based on detected and hypothesized planes. One of our major contributions, besides a treatment of data anisotropy and novel surface hypotheses, is a regularization of the reconstructed surface w.r.t. the length of edges and the number of corners. Compared to classical area-based regularization, it better captures surface complexity and is therefore better suited for man-made environments, such as buildings. To handle the underlying higher-order potentials, that are problematic for MRF optimizers, we formulate minimization as a sparse mixed-integer linear programming problem and obtain an approximate solution using a simple relaxation. Experiments show that it is fast and reaches near-optimal solutions.

Categories and Subject Descriptors (according to ACM CCS): I.2.10 [Artificial Intelligence]: Vision and Scene Understanding—3D/stereo scene analysis, I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Range data, I.5.4 [Pattern Recognition]: Applications—Computer vision

1. Introduction

Many applications make use of 3D models representing buildings and urban areas. It includes virtual navigation, urban planning, quantity survey as well as physical simulations for thermal performance, acoustics, lighting and shadow casting, pollutant dispersion, solar panel deployment. Qualitative applications, whose goal is to provide visual understanding, are generally based on detailed meshes with realistic rendering. On the contrary, quantitative applications call for a simplified geometry, that is better suited for simulation and for scaling up to large scenes. An increasing number of applications also require semantic information, e.g., building information models (BIM) or 3D city models (e.g., with CityGML). Creating this information, manually or automatically [BHMT13], is much easier on models with a simplified geometry. A market is in fact developing to offer model reconstruction services for existing buildings. It is based on the processing of dense 3D data obtained by laser scanners or photogrammetry. Model creation consists in manually adjusting geometric primitives to parts of the point cloud. It is labor intensive, error prone, and time and money consuming. Automatic building reconstruction has thus been an active field of research in the last years.

We present here a method for constructing a piecewise-planar approximation of an observed surface provided as range images. While most of our experiments are motivated by building and city reconstruction, the method is general and can be applied to any scene in man-made environments, where surfaces are mostly planar and where there are strong geometric priors, such as orthogonality and parallelism.

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

Challenges. One characteristic of buildings and urban environments is that, because of ubiquitous occlusions, many observations are required to see the whole scene. It goes beyond seeing all parts of concave areas and regions masked by pillars or open doors. Some areas are actually bound to remain hidden (e.g., behind or under furniture), to be hardly accessible to a device standing on a tripod (e.g., complete steps of a staircase), or to require many acquisitions (e.g., all intervals between exposed ceiling beams). A major challenge is thus the reconstruction of plausible surfaces for missing regions, to reduce the need for observations, and thus also to reduce the cost and time of data acquisition. We address it using robust plane hypotheses and an energy minimization with powerful higher-order regularization terms.

Another challenge is sampling anisotropy. Laser scans are often considered simple to treat as they have less noise and outliers compared to photogrammetric or Kinect-like data. They may however feature anisotropic data. Parts of the scene that are more distant from the scanner are less densely sampled than closer parts. Sampling of low-incidence surfaces also produces curved 3D lines of densely sampled points where the distance between lines can be one or two orders of magnitude larger than the distance between points on a line. Besides, for rotating laser scanners, regions close to poles of the sampling sphere are much more densely sampled than at the equator, typically with factors more than one thousand. Many approaches have robustness and accuracy problems with this issue. We address it by normalizing sampled data at each stage of the reconstruction process.

Related work. Many approaches have been proposed for reconstructing surfaces from point clouds with missing parts [BTS*14]. A number of them complete surfaces by inferring smooth geometry in areas with missing data [PBL10], which is inadequate for buildings and indoor scenes as they are mainly composed of planar regions and contain sharp features. Methods that handle these aspects first detect geometric primitives [SWK07, FP13], then reconstruct sharp features where primitives intersect. The choice of intersecting primitives is often based on proximity criteria and rely on the presence of observed points close to the intersection [LA13, CC08, LWC*11, JKS08]. These approaches are not suited for completing large regions with missing data when primitive intersections occur in hidden areas or far from observed points. User interaction might be needed in this case to select primitives and recover surfaces [CC08, ASF*13].

Only a few methods are able to complete large missing regions. In [Cas02], partially visible regions are extended until edges intersect, or intersect the wall or the floor. But this extension is made on a per-polygon basis, it does not prevent self-intersection and holes in the resulting surface. In [SDK09], a graph cut is performed on a graph aligned with a regular voxel grid favoring cuts close to detected primitives. However, it reconstructs only visible regions, possibly with artifacts due to voxel discretization. Other methods assume that all planes are oriented toward one of the three orthogonal dominant directions [FCSS09a, FCSS09b, BB10, VAB12]. This Manhattan-world assumption works well but is often too restrictive for indoor and detailed urban scenes. In [CLP10], planar regions can have long extensions in hidden areas. Detected planes are used to partition the 3D space into a polyhedral cell complex, and the reconstructed surface is defined as the interface of a volumic cell assignment (empty or full). It ensures a watertight surface without self-intersection. Moreover, to provide plausible completions in hidden areas, additional hidden plane hypotheses (called "ghosts") are considered. They are generated after regions are approximated by polygons, based on their edges. The idea is that, in a man-made environment, a straight edge of a planar region is likely to also be the edge of another planar region, often with a right angle. However, polygonization in [CLP10] is based on alpha shapes, which are sensitive to noise and sampling anisotropy, resulting in wrong or spurious plane hypotheses. More importantly, the prior used to regularize surfaces in hidden parts of the scene is area minimization, which is not an appropriate measure of surface simplicity for man-made environments.

One of our main contributions is the use of higher-order regularization terms that penalize the length of edges and/or the number of corners in the reconstructed surface. Such higher-order priors have been used for greedy mesh simplification but are challenging to incorporate in a global minimization for surface reconstruction. Note however that, contrary to a number of mesh simplifications approaches, we minimize the sum of the lengths of edges between non coplanar faces only, thus ignoring edges between coplanar faces. Therefore, our method ignores the problem of obtaining a good triangulation of each planar face and concentrate on the actual surface estimation problem. Our approach bares some similarity with some methods used for 2D image segmentation that also rely on higher-order regularization terms such as region boundary curvature [SKC09, SKR12]. In [SK11], a 3D surface is completed based on higher-order priors using a global binary labelization of a volume partition, similarly to us. However it requires a mesh as input and does not enforce visibility constraints available from range images. Besides, reconstruction quality is limited as it relies on a regular volume tessellation based on a few discrete plane directions. Last, it minimizes the mean curvature and produces a smooth surface, while we aim at reconstructing sharp features.

Our goal is not to reconstruct the most accurate surface, but to produce a simple piecewise-planar approximation at a given level of detail σ. Depending on the use, e.g., quantity survey or simulation, features such as baseboards, window ledges, roof tiles or chimneys may or may not be considered as useful pieces of information. This also matches with usual tolerances in the construction industry, e.g., an offset of 1 cm every 2 m for walls. To address the data anisotropy issue, a kind of information normalization is required so that the same choices are made in all regions of the scene, at any

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

distance and any incidence from the observation points. This normalization concerns both high-density areas and regions where the distance between points is higher than the scale of analysis. A standard way to handle anisotropy is to voxelize the point cloud [CLP10], which however has a number of drawbacks. First, it may introduce artifacts as the reconstructed surface may follow discrete voxel boundaries in addition to regression surfaces. Second, it introduces a new parameter, the voxel size, which can however be set to some value related to the requested level of detail, typically a fraction of the scale of analysis. Yet it only makes sense if not too small, as it rapidly introduces complexity issues. Last, it tends to introduce holes in regions whose density is low with respect to the scale of analysis; otherwise, voxel size is to be set higher, but then details of interest can be lost. Rather than voxelizing the scene, we directly normalize the influence of each 3D point, which prevents missing fragments in areas where the density is low compared to the scale of analysis.

Overview. Our approach can be summarized as follows:

- Planar primitives are detected in the point cloud, and possibly merged to recover from potential over-segmentation. The resulting planes are organized in an arrangement, representing plane hypotheses for the reconstructed surface.
- The edges of primitives are then extracted and classified. Edges considered as occluding create extra "ghost" plane hypotheses in the arrangement as scene priors. Ghosts parallel to primitives are generated too, that are well suited for thin flat objects without enough data on their border.
- Next, we associate to each cell of the complex a binary variable representing its occupancy status: a reconstructed surface is defined as the interface between empty and occupied cells. And we define an energy on cell occupancy that penalizes deviations to observations as well as structural complexity (to regularize invisible regions).
- Last, the corresponding discrete optimization problem is turned into an integer linear program. The final occupancy status is extracted from a solution to the relaxed problem, which experimentally proves to be relatively tight.

Our main contributions are the following:

- We present a thorough treatment of sampling anisotropy.
- We propose new schemes for generating plane hypotheses in hidden areas and for thin objects.
- We introduce new regularization priors that encode the length of edges and the number of corners in a reconstructed surface, possibly with angle preferences.
- We show how to efficiently minimize an energy based on these regularization terms.
- We illustrate on various experiments how these plane hypotheses and higher-order regularization terms significantly improve the quality of 3D reconstruction.

The paper is organized as follows. Section 2 describes plane hypotheses. Section 3 details the energy. Section 4 explains the optimization. Section 5 shows experimental validation.

2. Surface hypotheses

We formulate the surface reconstruction problem as an optimum binary labeling of the 3D space as empty or occupied. The reconstructed surface corresponds to the boundary of all the occupied volume. This ensures that the surface is watertight and without self-intersection. To make it amenable to a discrete optimization method with good optimality guaranties, we partition the 3D space a priori into a set of regions whose boundaries are plausible components of the surface.

2.1. Space partitioning

Two main partition methods have been used in the context of surface reconstruction: a regular voxel grid and a Delaunay triangulation of 3D points. The computational cost of a voxel grid can grow cubicly with the desired level of detail. It also has a bias in the estimation of surface areas and creates aliasing for surfaces not aligned on the grid. Moreover, it is hardly compatible with a sensible measure of edge length and corner count. In practice, because of noise in the data, Delaunay triangulations do not contain multi-face planes. [LA13] alleviates this issue by removing points near a detected plane and uniformly resampling it, which guarantees that the visible parts of the plane are covered by a set of faces in the triangulation. [vKvLV13] detect planar polygons and use a conforming constrained Delaunay triangulation to make sure polygons are preserved.

As in [CLP10], we partition the 3D space (within a bounding box) into a polyhedral cell complex using a plane arrangement structure. Each plane corresponds to a surface hypothesis. There are two kinds of hypotheses: planes detected as geometric primitives in the point cloud (cf. Section 2.2), and "ghosts", that are unobserved but possible planar surfaces associated to detected primitives (cf. Section 2.3). The arrangement is constructed by inserting each plane one after another. Each inserted plane cuts the volume into two half-spaces, resulting in a new set of cells, which are all convex as they correspond to the non-empty intersections of all half-spaces generated so far. We maintain adjacency information as we insert new planes, which allows an access to adjacent cell, edges and vertices in constant time. We denote $\mathcal{C}$ the final set of cells. It does not depend on insertion order. In contrast to [LA13, vKvLV13], this volume partition guarantees that planar primitives can be expanded far beyond their visibility area. It also allows the use of unseen plausible planes.

As the complexity of building such a plane arrangement is cubic in the number of planes in the worst case, it is better to restrict to planes that have good chances to be used in the reconstruction. The number of cells can be reduced too by limiting the extension of some planar primitives, using planes already in the arrangement as bounding surfaces. However, it creates a bias in plane insertion order. [CLP10] proposes two different strategies to limit planes. For aerial data, horizontal planes are inserted first; vertical planes are then inserted using horizontal planes as limiting planes; oblique planes are

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]



Figure 2: *Segmentation by region growing (left), plane fusion (middle) and line segments selected for ghosts generation.*

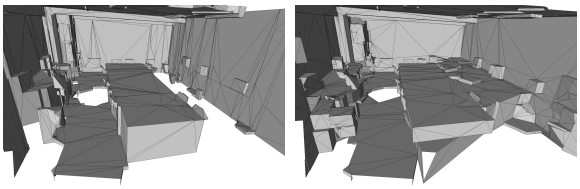


Figure 3: *Sensitivity to plane insertion order.*

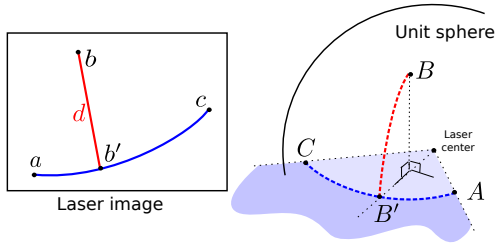


Figure 4: *Distance to line in range image for segment fusion.*

inserted last using both horizontal and vertical planes as limiting planes. For ground-level data, vertical planes are inserted before horizontal ones. We show in the experiment section that these strategies are often too aggressive and not suited for indoor scenes. Figure 3 illustrates the sensitivity to plane insertion order, showing widely different results for the two strategies. As described below, we not only have surface hypotheses different from [CLP10], we also use a less aggressive strategy for plane limitation in the arrangement.

2.2. Observed primitives

Many approaches have been proposed for detecting planes in a 3D point cloud, e.g., based on RANSAC [SWK07] or region growing. We exploit here the structure of the laser range images by segmenting the point cloud into planar regions using a region growing algorithm as in [PVBP08]. Points are first ordered according to their degree of planarity. A primitive is then grown iteratively from locally planar seed regions, adding neighboring points when close to the plane and with a similar normal. The plane is re-estimated each time the region is grown using an incremental estimate of the covariance. We estimate the normals of points with a robust method that preserves sharp features [BM12].

It differs from [CLP10] in that we consider as neighboring points the 8-pixel neighbors in the range image instead of the k-nearest neighbors in 3D. It is much faster and, more importantly, it prevent problems due to anisotropic data, in particular when laser sampling produces a series of (curved) lines of densely sampled points for high-incidence surfaces. This is not sensitive in the range image, but if region growing is based on k-nearest neighbors as in [CLP10], it tends to find all neighbors on a single line and thus to segment lines individually. And if k is increased, then robustness and accuracy are reduced for smaller primitives. In addition to [CLP10], we also perform plane fusion to recover from possible over-segmentation, based on efficient and robust statistical criteria for merging primitives [BM14] (see Figure 2).

2.3. Ghost primitives

We want to build a 3D cell complex whose facets include, for the visible parts of the scene, the detected primitives, and for invisible parts of the scene, plane hypotheses supported by the occluding edges of primitives. These ghost primitives are plausible surfaces that do not originate from direct observation, but that are likely enough given detected primitives and scene priors. In a man-made environment, an occluding straight edge of an observed planar primitive is likely to be the intersection with another, orthogonal plane. The generation of ghosts requires first the determination of primitive boundaries as well as their occlusion status.

One way to define the boundaries of a planar primitive defined from 3D points is to project the points on the support plane and to build their α-shape, as done in [CLP10] with $\alpha = \sigma^2$. As α-shapes tend to over-segment boundaries, post-processing is required to simplify the polygonal contours, such as greedy iterative edge merging [CLP10]. But a major drawback of using alpha shapes to define boundaries is that points are disconnected from an α-shape as soon as their distance to other point of the primitives is larger than $\sqrt{\alpha}$, resulting in truncated primitives. It occurs typically for distant planes with high incidence such as ceilings and floors.

To prevent it, we consider boundaries in the range image, i.e., pixel chains of primitive contours, discarding small regions for robustness to noise and clutter. The primitive edges are then given by a simplification of the contour lines that abstract from the aliasing (discretization) effect, which is dominant here w.r.t. noise, yielding a polygon with holes. For this, we iteratively merge two adjacent segments ab and bc into segment ac if the distance of b to ac is below a fixed pixel threshold δ, starting with points with the smallest distance. With a spherical acquisition, straight 3D lines appear as curves in the range image. To measure the deviation of a 2D point b with respect to a 2D line ac, we consider their projection A, B, C on the unit sphere (see Figure 4). Let B' be

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

the geodesic projection of B on the geodesic line AC, and let b' be the projection of B' on the range image. Segments ab and bc are merged into ac if the distance between b and b' (rather than ac) is below the fixed pixel threshold δ. To make sure that edges do not drift, we actually keep track of all points merged into lines segments and ensure at each iteration that new segments stay within the same pixel distance δ to the original contour. The Haussdorf distance between the original pixel contour and the simplified polygon, adjusted according to spherical projection, is thus also bounded by δ.

Projecting the 2D segments onto their 3D support planes, we obtain bounding 3D segments for each primitive plane. There are three kinds of segments: adjacency segments (at the visible intersection of two observed primitives), occluding segments (when the primitive boundary is significantly in front of what lies on the other side of the segment) and occluded segments (when the primitive boundary is significantly behind what lies on the other side of the segment). Generating a ghost for adjacency segments is not necessary, if not inaccurate or wrong. As the surface at occluded segments is likely to continue behind what occludes it, we generate ghost planes for occluding segments only (see Fig. 2). For robustness to anisotropy, we identify occluding segments as segments such that (1) a majority of segment points in the range image lies close to points behind the plane, and (2) the segment is not an adjacency segment, which excludes salient edges. Two primitives are considered as adjacent if at least two points, one in each projection of a primitive on its support plane, are mutual neighbors, and a segment is an adjacency segment if it intersects in the range image the reprojected line of intersection of the two 3D planes.

In [CLP10], ghosts are generated for each edge of the polygonal contour of each primitive, unless the edge is considered as adjacent to another primitive, i.e., at a distance less than the observation scale σ. Compared to us, this systematic generation of ghosts for all edges (except adjacency edges) results in many spurious ghosts for edges of occluded primitives, on the order of twice too many. It greatly reduces speed and scalability because of the cubic complexity of plane addition into the arrangement. It also reduces accuracy by introducing dummy planes that can be chosen when the surface is reconstructed. Note however that deciding whether a point on the edge of a primitive occludes another primitive is difficult with unstructured 3D point sets, especially when there are several visibility points and density variations.

Finally, typical indoor scenes also contain thin plane-parallel objects for which only one plane is visible, e.g., an open door or a table top, whose edges are too thin to be detected as planar primitives. To reconstruct such an object, we need to generate a plane hypothesis for the hidden opposite plane. It requires an estimation of the object thickness. For this, we extrude the planar region in the direction opposite to its normal, trying various thickness hypotheses and checking that the extruded volume does not intersect with the visibility cone, i.e., rays from the scanner center to observed points. Thicknesses are tried with increasing values based on a constant increment, thus yielding a maximum thickness. For robustness to noise, we include some tolerance when testing ray intersection and consider a thickness as too large only if a minimum number of intersecting rays are found.

3. Surface reconstruction

We want to reconstruct a surface that mostly conforms to observations in the visible planar parts of the scene, that does not intersect the lines of sight of observed points, and that is plausible and simple in hidden parts.

We formulate the surface reconstruction problem as an optimal binary labeling (occupied or empty) of each 3D cell of the complex defined in Section 2. The surface is given by the set of facets that lie on the interface between empty and occupied regions, i.e., facets adjacent to both an empty and an occupied cell. This guarantees watertightness and the absence of self-intersections. To represent the occupation assignment, we associate a discrete variable $x_c \in \{0,1\}$ to each cell $c \in \mathcal{C}$, where 0 represents an empty cell and 1 an occupied cell. We define the energy of an occupation assignment $\mathbf{x} = (x_c)_{c\in\mathcal{C}}$ as composed of two terms, a data term and a regularization term:

$$E(\mathbf{x}) = E_{\mathsf{data}}(\mathbf{x}) + E_{\mathsf{regul}}(\mathbf{x}) \tag{1}$$

The regularization term acts as a prior to favor plausible surfaces of man-made environments, mainly in hidden regions.

Notations. Each point $p \in \mathcal{P}$ belonging to a primitive Π_p based on plane P_p is assigned to a facet f_p of the cell complex, given by the projection of p on P_p. The assignment of points to facets is done when the plane is put in the arrangement and updated as more planes are inserted and facets are split. The subscript is dropped when unambiguous. Given a facet $f \in \mathcal{F}$ in the complex, we note f^+ and f^- the sides of f corresponding respectively to the positive (i.e., visible) and negative (i.e., hidden) sides of the underlying plane (primitive or ghost). We also note c_{f^+} and c_{f^-} the corresponding cells on both sides of f, and $x_{f^+}, x_{f^-} = x_{c_{f^+}}, x_{c_{f^-}}$.

3.1. Data terms

To mostly conform to visible features of the scene, the data term penalizes different kinds of deviations from observations. It is composed of two terms, a primitive term $E_{\mathsf{prim}}(\mathbf{x})$ and a visibility term $E_{\mathsf{vis}}(\mathbf{x})$:

$$E_{\mathsf{data}}(\mathbf{x}) = E_{\mathsf{prim}}(\mathbf{x}) + E_{\mathsf{vis}}(\mathbf{x}) \tag{2}$$

Primitive term. $E_{\mathsf{prim}}(\mathbf{x})$ penalizes primitive points that are not on the reconstructed surface, with proper orientation. A penalty is given to point p if its corresponding facet f is not part of the reconstructed surface with appropriate orientation, i.e., if the cell on the positive side of f is not empty and if the cell on the negative side of f is not occupied.

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

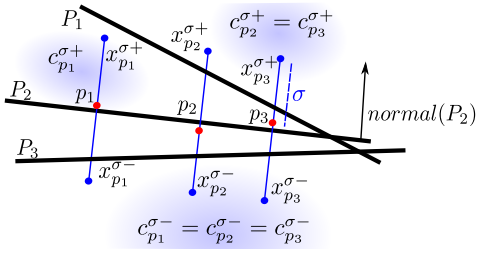


Figure 5: *Tolerance for the points-on-primitive penalty.*

As there may be some measurement error in the position of p and in the detection of the underlying plane P_p, we actually add some tolerance around P_p. We consider the facets intersected by a line segment of length σ, orthogonal to P_p, and starting at the orthogonal projection of p on P_p, on both sides of P_p (see Figure 5). Let $f_p^{\sigma+}$ be the facet whose intersection point with this line segment is on the positive side of P_p, is at most at distance σ to P_p, and is the most distant from P_p. (There is at least one such facet: f itself.) Symmetrically, let $f_p^{\sigma-}$ be the facet with the same definition on the negative side of P_p. Let $c_p^{\sigma+}$ and $c_p^{\sigma-}$ be the cells on the positive and negative sides of $f_p^{\sigma+}$ and $f_p^{\sigma-}$, and $x_p^{\sigma+}, x_p^{\sigma-}$ be the corresponding variables. The penalty is defined as the sum of $x_p^{\sigma+}$ and $1-x_p^{\sigma-}$, which should both be 0. Note that the cells between $c_p^{\sigma+}$ and $c_p^{\sigma-}$, if any, are ignored. Imposing a single empty-to-occupied transition on the line of sight around p is not necessary in practice thanks to regularization.

To take sampling anisotropy into account, we assign a different weight to each observation point. The weight measures the area of the reconstructed surface that does not comply to observations, relatively to the scale. It is adimensional, homogeneous to a number of area units w.r.t. scale, i.e., a number of times σ^2. It is related both to the local density of observation rays, which is higher near the pole, and to their incidence on planes. We consider here a rotating 3D laser scanner that acquires a spherical view of the surrounding scene with measures taken on a sequence of vertical planes. Data points are given by an azimuth angle $\theta \in [0, 2\pi[$ in the horizontal plane and a polar angle $\phi \in [0, \pi]$ (measured from the zenith direction) in the corresponding vertical plane. Sampling discretizes the range of angles with constants steps Δ_θ and Δ_ϕ. We consider a point p, acquired in direction (θ, ϕ) and observed at distance d on a plane P with incidence angle $\psi \in [0, \pi/2[$ w.r.t. the observation ray. We give it a weight that is the relative area of the projection on P of the patch on the unit sphere corresponding to the angular discretization step $\Delta_\theta \times \Delta_\phi$, made relative to the polar angle to cope with the variation of sampling density:

$$w_p^{\mathsf{aniso}}(P) = \frac{d^2}{\sigma^2} \Delta_\theta \Delta_\phi \frac{\sin\phi}{\cos\psi} \tag{3}$$

Finally, the actual primitive term is as follows:

$$E_{\mathsf{prim}}(\mathbf{x}) = \sum_{p \in \mathcal{P}} w_p^{\mathsf{aniso}}(P_p) \left(x_p^{\sigma+} + (1 - x_p^{\sigma-}) \right) \tag{4}$$

Visibility term. $E_{\mathsf{vis}}(\mathbf{x})$ penalizes reconstructed surfaces between observed points and the sensor. For each point p, we consider the line of sight between p and its observation point ω (the laser center). There should be no matter, hence no reconstructed surface, on the ray ωp. For that, we forbid the cell containing ω to be occupied (with an infinite cost) and we penalize facets that are transitions between occupied and empty cells along observation rays. A penalty is paid each time an interface is traversed, from inside or outside.

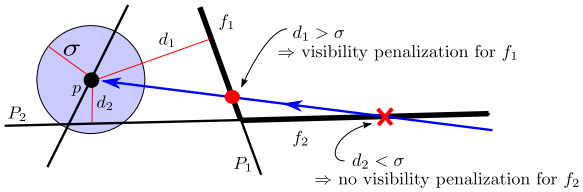


Figure 6: *Tolerance for the visibility penalty.*

Like for the primitive term, to be robust to measurement errors in the position of p, we actually add some tolerance and do not penalize facets f whose plane P_f is at most at distance σ from p (see Figure 6). As for the primitive term too, we take sampling anisotropy into account and weigh the penalty of an interface facet f by the relative area of the sampling ray on the surface interface, i.e., $w_p^{\mathsf{aniso}}(P_f)$. However, contrary to the primitive term, all observed points are taken into account here, whether they belong to a primitive or not:

$$E_{\mathsf{vis}}(\mathbf{x}) = \sum_{\substack{p \in \mathcal{P},\ f \in \mathcal{F} \\ \omega p \cap f \neq \emptyset,\ d(p,P_f) \leq \sigma}} w_p^{\mathsf{aniso}}(P_f) \left| x_{f^+} - x_{f^-} \right| \tag{5}$$

Compared to [CLP10], we separate the penalty of observed points into primitive reconstruction requirements and visibility issues. Also, for visibility, we penalize all planes, not only reconstructions relying on the negative side of a plane. More importantly, [CLP10] counts a constant cost for all disagreeing observed points, whereas we weigh penalties depending on point density. It also provides a unit common ground to balance $E_{\mathsf{vis}}(\mathbf{x})$ and $E_{\mathsf{prim}}(\mathbf{x})$ into a single term.

3.2. Regularization terms

As we want to reconstruct the simplest plausible surface in hidden parts of the scene, regularization terms penalize surface complexity. Note that surface regularization applies to the whole scene, not just the visible part. Yet, the weight of regularization terms is lower than the weight of data terms to make sure that the reconstruction mostly conforms to observations, unless a few observed points can be considered as outliers and traded for a greater surface simplicity.

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

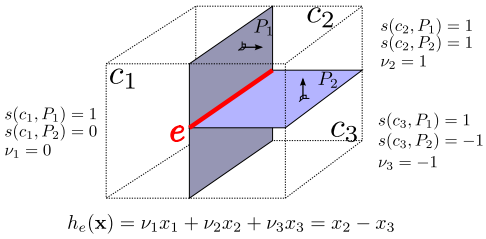


Figure 7: *Computation of weights $\nu(c,e)$ and edge measure function $h_e(\mathbf{x})$ in the case of 3 adjacent cells.*

We introduce three regularization terms, that can be used separately, or jointly with relative weights: an area term, an edge term and a corner term:

$$E_{\text{regul}}(\mathbf{x}) = \lambda_{\text{area}} E_{\text{area}}(\mathbf{x}) + \lambda_{\text{edge}} E_{\text{edge}}(\mathbf{x}) + \lambda_{\text{corner}} E_{\text{corner}}(\mathbf{x}) \tag{6}$$

Area term. $E_{\text{area}}(\mathbf{x})$ penalizes the area of the reconstructed surface, i.e., the sum of the areas a_f of facets f separating an empty cell from an occupied one, relatively to scale σ:

$$E_{\text{area}}(\mathbf{x}) = \sum_{f\in\mathcal{F}} w_f\,|h_f(\mathbf{x})| \quad \text{where} \quad \begin{cases} w_f = a_f/\sigma^2 \\ h_f(\mathbf{x}) = x_{f^+} - x_{f^-} \end{cases} \tag{7}$$

It can be seen as a sum of submodular pairwise potentials in the context of Markov Random Fields (MRF). This kind of term has been widely used in surface reconstruction methods based on energy minimization [CLP10].

Edge term. $E_{\text{edge}}(\mathbf{x})$ penalizes the length of edges in the reconstructed surface. An edge $e \in \mathcal{E}$ in the cell complex lies on the intersection line of two non-parallel planes, with end points at the intersection with two other planes in the complex. It has 4 adjacent cells in general, but may have only 3 as ghosts just introduce half-planes in the arrangement; one cell then lies alone on one side of a plane (see Figure 7). To model both cases, we introduce a function s that associates a sign to each cell-plane pair:

$$s(c,P) = \begin{cases} +1 \text{ if } c \text{ is on the positive side of } P \\ -1 \text{ if } c \text{ is on the negative side of } P \\ 0 \text{ if } c \text{ is on both sides of } P \end{cases} \tag{8}$$

An edge in the complex that has 4 adjacent cells is an actual "material" edge of the reconstructed surface if and only if one only of the 4 cells is empty (re-entrant edge) or one only is occupied (salient edge), see Figure 8. An edge that has 3 adjacent cells is a material edge iff the two cells lying on the same side of the plane have different values. In both cases, the existence of a material edge at e can be expressed by:

$$h_e(\mathbf{x}) = \sum_{c\in\text{Adjc}(e)} \nu(c,e)\,x_c \quad \text{where} \quad \nu(c,e) = \prod_{P\in\text{AdjP}(e)} s(c,P) \tag{9}$$

$\text{Adjc}(e)$ is the set of cells adjacent to e and $\text{AdjP}(e)$ the set of adjacent planes, which are accessed in constant time (cf. Section 2.1), and $\nu(c,e)$ a sign used as a linear weight associated to each adjacent cell (see Figure 8). The absolute value $|h_e(\mathbf{x})|$ is equal to 0 when there is no material edge, to 1 when there is 1 edge (re-entrant or salient) and to 2 when there are 2 re-entrant and 2 salient edges. It is independent on the convention for assigning a sign to the supporting planes.

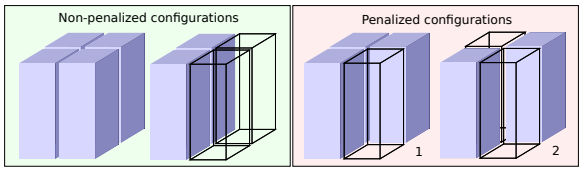


Figure 8: *Possible configurations of the 4 cells adjacent to an edge and corresponding penalization. Other configurations can be obtained by rotation and occupancy inversion.*

This term corresponds in the 4-cell case to a 4th-order potential in the context of MRFs, and in the 3-cell case to a 2nd-order potential as one variable then has a null factor. It can be shown that this 4th-order potential cannot be made regular in Kolmogorov and Zabih's sense [KZ04]. We actually penalize edges in the reconstructed surface by:

$$E_{\text{edge}}(\mathbf{x}) = \sum_{e\in\mathcal{E}} w_e\,|h_e(\mathbf{x})| \tag{10}$$

where w_e is a weight associated to e. This weight is made proportional to the edge length l_e to make the total length of material edges independent of plane splitting in the complex. It is made relative to scale σ too. It can also reflect the inadequacy of the angle α_e between the two planes of e with respect to an expected angle distribution. We penalize here angles far from 90°, using two parameters: the cost A of having a angle α in the scene widely different from a right angle, and the expected standard deviation ρ of an angle w.r.t. the right angle. The angular weight $w_{\text{ang}}(\alpha)$ is defined as:

$$w_{\text{ang}}(\alpha) = A + (1-A)\exp\left(-\frac{(\alpha-\pi/2)^2}{2\rho^2}\right) \tag{11}$$

The weight $w_{\text{ang}}(\alpha)$ is equal to 1 if $\alpha = \pi/2$, and rapidly reaches A as soon as α departs from $\pi/2$ with an offset more than ρ. Finally, we define the edge weight w_e as:

$$w_e = \frac{l_e}{\sigma}\, w_{\text{ang}}(\alpha_e) \tag{12}$$

Corner term. $E_{\text{corner}}(\mathbf{x})$ penalizes the number of corners in the reconstructed surface. A vertex $v \in \mathcal{V}$ in the complex is the point of intersection of three non-parallel planes. It has 8 adjacent cells in general, but can have only 6 or 4 as ghosts introduce half-planes. The existence of an actual "material" corner in the reconstructed surface can be expressed by:

$$h_v(\mathbf{x}) = \sum_{c\in\text{Adjc}(v)} \nu(c,v)\,x_c \quad \text{where} \quad \nu(c,v) = \prod_{P\in\text{AdjP}(v)} s(c,P) \tag{13}$$

$\nu(c,v)$ is a sign used as a linear weight associated to each adjacent cell (see Figure 9). $|h_v(\mathbf{x})|$ is equal to 0 when there is no material corner, to 1 when there is a single solid or empty corner, and can rise up to 4 in complex corner cases.

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

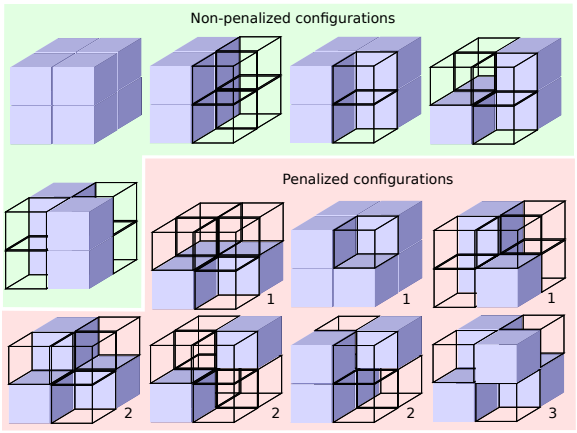


Figure 9: *Possible configurations for the 8 cells adjacent to a corner and their corresponding penalization. Penalized configurations are in the red zone. Other configurations can be obtained by rotation, symmetry and occupancy inversion.*

Note that the saddle point (4th case on the first line in Figure 9) is not penalized. This can be justified from a complexity point of view: assuming that putting two cuboids in contact should not be measured as more complex than two separate cuboids, the saddle points that it can create (up to 4) should not be penalized. As for edges, these terms do not depend on the convention for assigning a sign to the supporting planes. They correspond to potentials of order up to 8 in the context of MRFs. We actually penalize corners with:

$$E_{\text{corner}}(\mathbf{x}) = \sum_{v \in \mathcal{V}} w_v \, |h_v(\mathbf{x})| \tag{14}$$

where w_v is a weight associated to v. Indeed, we may want to discourage 3D corners that do not feature right angles. We consider the three angles $(\alpha_{v,i})_{i\in\{1,2,3\}}$ between each pair of plane passing through v. As with edges, the corner penalty is then made proportional to the following weight:

$$w_{\text{ang}}(\alpha_1,\alpha_2,\alpha_3) = A + (1-A)\exp\left(-\frac{\sum_{i\in\{1,2,3\}}(\alpha_i - \pi/2)^2}{2\rho^2}\right) \tag{15}$$

It is independent of plane order. Finally we define:

$$w_v = w_{\text{ang}}(\alpha_1,\alpha_2,\alpha_3) \tag{16}$$

4. Optimization

We estimate the surface to reconstruct through the minimization of energy $E(\mathbf{x})$. It could be done using a discrete optimization method based on an MRF formulation with higher-order potentials, up to 8th-order for the corner term. However such higher-order terms are known to be challenging for most MRF inference techniques. Efficient graph-cut methods can be used if we have submodular pairwise potentials, which is the case only if we do not use the edge and corner terms (as in [CLP10]), but they are not suited to models that contain such higher-order non-submodular potentials. We tested various methods that can handle higher-order potentials in the OpenGM library [KAH*13], including tree-reweighted belief propagation and lazy flipper, without much success in terms both of running time and quality of the found optimum.

Mixed-integer programming formulation. Instead of using an MRF formulation, we express the problem as a linear integer program. The main idea is to reformulate each absolute value in each term of the energy as a linear program using a continuous auxiliary variable as we have:

$$|x| = \min_y y \quad s.t. \quad -y \le x \le y \tag{17}$$

Based on this idea, we introduce a set of continuous auxiliary variables $\mathbf{y}$ for each facet, edge and corner, and a set of linear constraints on these variables:

$$(\mathbf{x},\mathbf{y}) \in \mathscr{C} \Leftrightarrow \begin{cases} \forall f \in \mathcal{F}, & -y_f \le h_f(\mathbf{x}) \le y_f \\ \forall e \in \mathcal{E}, & -y_e \le h_e(\mathbf{x}) \le y_e \\ \forall v \in \mathcal{V}, & -y_v \le h_v(\mathbf{x}) \le y_v \end{cases} \tag{18}$$

We can then reformulate the minimization $E(\mathbf{x})$ as the minimization of a linear function of both sets of variables $\mathbf{x}$ and $\mathbf{y}$, which becomes a mixed-integer linear program of the form:

$$E'(\mathbf{x},\mathbf{y}) = \zeta + \sum_{c\in\mathcal{C}} w'_c x_c + \sum_{f\in\mathcal{F}} w'_f y_f + \sum_{e\in\mathcal{E}} w'_e y_e + \sum_{v\in\mathcal{V}} w'_v y_v \tag{19}$$

$$\min_{\mathbf{x}} E(\mathbf{x}) = \min_{\mathbf{x},\mathbf{y}} E'(\mathbf{x},\mathbf{y}) \quad s.t. \begin{cases} \forall c \in \mathcal{C} \;\; x_c \in \{0,1\} \\ (\mathbf{x},\mathbf{y}) \in \mathscr{C} \end{cases} \tag{20}$$

Optimization using LP-Relaxation. Solving a integer program is NP-Hard in general. We take the classical approach that consists in relaxing the problem by allowing each variables x_c to take values in the interval $[0,1]$. This leads to a linear programming (LP) problem that can be solved efficiently using of-the-shelf solvers. The method based on the dual simplex in Mosek© has been successfully used in our experiments. In case we only use area penalization, all terms are submodular, the relaxation is tight, and solving the relaxed problem yields an integral solution that is the global optimum of the corresponding integer programming problem. When using the higher-order regularization terms, we obtain fractional values that have to be rounded, yielding a suboptimal integral solution. We use the most simple rounding strategy that consists in rounding each primary variable x_c independently of other variables. After this rounding step, we solve the linear program again by constraining each primary variables x_c to remain fixed to the rounded value. It re-estimates the slack variables given the chosen solution of the primary variables, and thus estimates the increase of the objective function caused by the rounding. We observed experimentally that this energy increase is small in general: up to 8% when using only corner regularization and up to 6% with edge regularization, depending on scene complexity. It shows we find solutions that are almost as good as the

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

global optimum. We also experimented with the minimization of the original integer programming problem using the "branch & bound & cut" algorithm implemented in Mosek© and obtained running times that where much longer for only a marginal improvement of the reconstructed surface.

5. Experiments

We evaluated our surface reconstruction method on interior scenes sampled with a 3D laser scanner. We compared to the general piecewise-planar reconstruction method in [CLP10], at the same scale σ. (Comparison only makes sense with methods that feature surface completion in hidden area and piecewise-planar priors. We thus did not compare with Poisson or other smooth reconstruction approaches because they are not appropriate for the kind of scene we target and because they produce a complex geometry that is not suited for quantitative applications. We did not compare either with approaches based on a Manhattan-world assumption, which present obvious artifacts.)

For all the experiments reported here, we used $\sigma = 10\,\text{cm}$. When using one kind of regularizer only, we defined $\lambda_{\text{area}} = 10^{-4}$, $\lambda_{\text{edge}} = 10^{-3}$ and $\lambda_{\text{corner}} = 10^{-2}$. When using a combination of edge and corner regularization, we defined $\lambda_{\text{edge}} = 5 \times 10^{-4}$ and $\lambda_{\text{corner}} = 10^{-2}$. The idea is that we want first of all to favor corner minimization, but that between two solutions with equal or similar number of corners, the one with the minimum number of edges should be preferred: 1 corner can only be traded for $\sigma\lambda_{\text{corner}}/\lambda_{\text{edge}} = 2\,\text{m}$ of edges. It somehow acts as a kind of approximate lexicographical order. These coefficients make regularization negligible compared to the data term, which stays preeminent.

As shown on Figure 1 (1st image), the parts of the scene that are distant from the observation point are not well reconstructed with [CLP10], due to anisotropy sensitivity. Playing with the scale σ of [CLP10] does not improve results, but on the contrary degrades them: if smaller, distant parts of the scene are not reconstructed; if larger, details are lost. On the contrary, our method is robust to the anisotropy of sampling. However, area regularization still introduces unwanted effects, such as a hole on the ground in the invisibility cone under the laser tripod (2nd image on Figure 1), which happens to reduce the area of the reconstructed surface. Regularizations with edges and corners (3rd and 4th image) are much better as leaving a hole in this invisible region is now penalized because of useless edges or corners.

There is only a marginal difference between corner and edge regularization in Figure 1. But Figure 10 provides an example where corner minimization is clearly better than edge minimization. Yet, both regularizers used jointly can be superior to corner regularization alone, as can be seen in Figure 11. Figures 12 and 13 provide other examples. Despite a few errors due to clutter in the original scene, the reconstruction is mostly correct, including for thin objects (e.g., tables, window jambs), whose edges are too small to be detected as planar primitives. Note that the images are seen from a different viewpoint than the observation point. For instance, the laser scanner was to low to acquire any data of the treads on the top half of the stairway, and could only sample parts of the risers. Still, a full staircase is reconstructed.

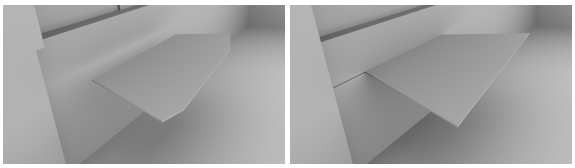


Figure 10: *Edges (left) vs corners (right).*



Figure 11: *Corners (left) vs edge+corner (right).*

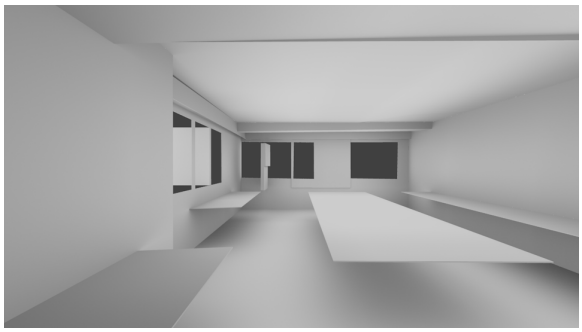


Figure 12: *Meeting room 2 with corner regularization.*



Figure 13: *Stairs scan and corner regularization.*

Some execution times are provided in Table 1. The total running time is a few minutes. Although our system could handle a laser acquisition with tens of millions of points, as provided by full-resolution scans, we found it effective enough to work on downsampled images (with a factor 25). Indeed, the bottleneck is the time to construct the arrangement, which is independent of the number of points. The number of variables in the linear program also does not depend on the number of points, although the weights add up the information of each data point, with a cost of intersecting each line of sight with the arrangement.

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]

Scene	#points	Normals	Detection		Fusion		Ghost generation		Arrang.	Optimization		Total
Meeting room 1 st. 1	658 k	18 s	0 s	72 prim.	2 s	39 prim.	27 s	344 planes	54 s	corner	52 s	153 s
Meeting room 1 st. 2	642 k	24 s	1 s	48 prim.	1 s	27 prim.	23 s	211 planes	57 s	corner	50 s	156 s
Meeting room 1 st. 3	663 k	22 s	1 s	52 prim.	1 s	26 prim.	23 s	276 planes	32 s	corner	16 s	95 s
Meeting room 1 st. 4	667 k	23 s	2 s	56 prim.	1 s	26 prim.	24 s	230 planes	52 s	corner	42 s	144 s
Meeting room 2	1054 k	49 s	2 s	36 prim.	1 s	21 prim.	34 s	232 planes	64 s	corner	18 s	168 s
Stairs	680 k	18 s	0 s	51 prim.	1 s	40 prim.	26 s	160 planes	47 s	corner	45 s	137 s

Table 1: *Computation time of surface reconstruction stages for various scenes.*

6. Conclusion

We have presented an effective method for reconstructing surfaces from range images, that infers plausible completions in hidden regions using priors adapted to man-made environments. Compared to [CLP10], which we build on and improve in many respects, it has no artifact due to voxel discretization nor to the order of plane insertions in the arrangement, and it handles sampling anisotropy and thin objects. Moreover, as shown by our experiments, our edge and corner regularizations are unquestionably superior to area minimization to reconstruct plausible surfaces in hidden regions. Our formulation for these regularizers is solved efficiently by LP relaxation, reaching near-optimal global solutions. Note that these regularizers and their optimization are not restricted to range images; they can be used in other contexts, e.g., to treat photogrammetric data as in [CLP10].

Although we can treat a laser scan with millions of points, we cannot process hundreds of such scans to reconstruct a whole building. A challenge now is to efficiently combine multi-view partial reconstructions while ensuring geometric consistency and a global (near-)optimum. Semantizing the reconstructed geometry also is a major issue.

References

[ASF*13] ARIKAN M., SCHWÄRZLER M., FLÖRY S., WIMMER M., MAIERHOFER S.: O-snap: Optimization-based snapping for modeling architecture. *TOG* (2013). 2

[BB10] BUDRONI A., BÖHM J.: Automatic 3D modelling of indoor Manhattan-world scenes from laser data. In *ISPRS Symp. Close Range Image Measurement Techniques* (2010). 2

[BHMT13] BOULCH A., HOULLIER S., MARLET R., TOURNAIRE O.: Semantizing complex 3D scenes using constrained attribute grammars. *Computer Graphics Forum* (2013). 1

[BM12] BOULCH A., MARLET R.: Fast and robust normal estimation for point clouds with sharp features. *CGF* (2012). 4

[BM14] BOULCH A., MARLET R.: Statistical criteria for shape fusion and selection. In *ICPR* (2014). 4

[BTS*14] BERGER M., TAGLIASACCHI A., SEVERSKY L. M., ALLIEZ P., LEVINE J. A., SHARF A., SILVA C.: State of the art in surface reconstruction from point clouds. *Eurographics STAR (Proc. of EG'14)* (2014). 2

[Cas02] CASTELLANI U.: Improving environment modelling by edge occlusion surface completion. In *3DPVT* (2002). 2

[CC08] CHEN J., CHEN B.: Architectural modeling from sparsely scanned range data. *IJCV* (2008). 2

[CLP10] CHAUVE A.-L., LABATUT P., PONS J.-P.: Robust piecewise-planar 3D reconstruction and completion from large-scale unstructured point data. In *CVPR* (2010), pp. 1261–1268. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

[FCSS09a] FURUKAWA Y., CURLESS B., SEITZ S., SZELISKI R.: Manhattan-world stereo. In *CVPR* (2009). 2

[FCSS09b] FURUKAWA Y., CURLESS B., SEITZ S., SZELISKI R.: Reconstructing building interiors from images. In *ICCV* (2009), pp. 80–87. 2

[FP13] FAYOLLE P.-A., PASKO A.: Segmentation of discrete point clouds using an extensible set of templates. *Vis. Comput. 29*, 5 (2013), 449–465. 2

[JKS08] JENKE P., KRÜCKEBERG B., STRASSER W.: Surface Reconstruction from Fitted Shape Primitives. In *VMV* (2008). 2

[KAH*13] KAPPES J. H., ANDRES B., HAMPRECHT F. A., SCHNÖRR C., NOWOZIN S., BATRA D., KIM S., KAUSLER B. X., LELLMANN J., KOMODAKIS N., ROTHER C.: A comparative study of modern inference techniques for discrete energy minimization problem. In *CVPR* (2013). 8

[KZ04] KOLMOGOROV V., ZABIH R.: What energy functions can be minimized via graph cuts? *PAMI* (2004). 7

[LA13] LAFARGE F., ALLIEZ P.: Surface reconstruction through point set structuring. *CGF* (2013). 2, 3

[LWC*11] LI Y., WU X., CHRYSANTHOU Y., SHARF A., COHEN-OR D., MITRA N. J.: GlobFit: consistently fitting primitives by discovering global relations. *TOG* (2011). 2

[PBL10] PAULSEN R. R., BAERENTZEN J. A., LARSEN R.: Markov random field surface reconstruction. *TVCG 16* (2010). 2

[PVBP08] POPPINGA J., VASKEVICIUS N., BIRK A., PATHAK K.: Fast plane detection and polygonalization in noisy 3D range images. In *IROS* (2008), pp. 3378–3383. 4

[SDK09] SCHNABEL R., DEGENER P., KLEIN R.: Completion and reconstruction with primitive shapes. *CGF* (2009). 2

[SK11] STRANDMARK P., KAHL F.: Curvature regularization for curves and surfaces in a global optimization framework. In *EMMCVPR* (2011), pp. 205–218. 2

[SKC09] SCHOENEMANN T., KAHL F., CREMERS D.: Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *ICCV* (2009). 2

[SKR12] SHEKHOVTSOV A., KOHLI P., ROTHER C.: Curvature prior for MRF-based segmentation and shape inpainting. In *DAGM/OAGM Symposium* (2012), vol. 7476 of *LNCS*. 2

[SWK07] SCHNABEL R., WAHL R., KLEIN R.: Efficient RANSAC for point-cloud shape detection. *CGF* (2007). 2, 4

[VAB12] VANEGAS C. A., ALIAGA D. G., BENES B.: Automatic extraction of Manhattan-world building masses from 3D laser range scans. *IEEE TVCG* (2012). 2

[vKvLV13] VAN KREVELD M. J., VAN LANKVELD T., VELTKAMP R. C.: Watertight scenes from urban LiDAR and planar surfaces. *Comput. Graph. Forum* (2013), 217–228. 3

This is the accepted version of the following article: *Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization* Alexandre Boulch, Martin de La Gorce and Renaud Marlet Symposium of Geometry Processing 2014 Computer Graphics Forum, which has been published in final form at [http://onlinelibrary.wiley.com/doi/10.1111/cgf.12431/abstract]