Robust Reconstruction of Watertight 3D Models from Non-uniformly Sampled Point Clouds without Normal Information

1. Briefly summarize the paper's contributions. Does it address a new problem? Does it present a new approach? Does it show new types of results?

- **[AS]**
  This paper presents a new approach to the problem of surface reconstruction from a 3D point cloud without normal information. Unlike many other methods that use a signed distance function to extract the surface, this paper uses an unsigned distance function to avoid the problems that signed distance functions inevitably face in the presence of noise in the normal estimation of the point cloud. This algorithm estimates a pseudo distance function in a subset of the voxel grid near the input points, and uses this distance function of compute the minimum cut of graph embedded within the voxels. Once they obtain this minimum cut, they present a method to convert it into a closed, manifold triangle mesh with a minimal number of vertices.
- **[DS]**
  The paper presents a volumetric method of reconstructing a watertight triangle mesh from an unoriented point clouds. It produces meshes of low genus, which other methods that use signed distance function do not.
  The new approach presented uses an unsigned distance function so that surface orientation is not required for the computation (normals are not needed).
  A hierarchical approach is used to enable efficient reconstruction of noisy and irregular data and prevent the loss of fine details.
  Good results are obtained when running this method on non-uniform point clouds with significant noise, outliers and holes.
- **[FP]**
  The paper presents a technique for surface reconstruction of unoriented points based on hierarchical segmentation of a voxelized structure. As in Kolluri *et. al.* paper, the segmentation approach formulates the surface reconstruction as a global optimization problem. The authors claim that one of the main novelties of their approach is introducing a confidence function comparable to an unsigned distance function. They assert that this kind of unsigned distance makes their reconstruction more robust to topological artifacts. This topological robustness is also supported by the graph-cut segmentation that tends to produce just few disconnected components.
- **[JD]**
  The paper presents a method of watertight surface reconstruction using the min-cut algorithm on a graph with an unsigned distance function. The algorithm is an improvement on previous signed distance function algorithms because it does not require normals, it can handle noise, and it produces meshes with low genus.
- **[LF]**
  The paper presents a method which reconstructs a surface based on a volumetric unsigned distanced function. The method is able to avoid the problem of acquiring accurate local surface orientation data, a problem inherent to methods which use signed distance functions, and in turn is robust to noise and non-uniform distribution of samples.
- **[MK]**
  The approach formulates the surface reconstruction problem as a graph-cut problem. However, unlike previous methods, the nodes of the cut graph do not correspond to volumes of a space partition but rather to the faces of a partition. None-the-less the

authors show (or rather rely on their earlier work that same year to show) that the resulting mesh is water-tight.

2. What is the key insight of the paper? (in 1-2 sentences)

- **[AS]**
  The key insight of this paper is that using an unsigned distance function can produce highly accurate surface reconstructions without the need for normals and without facing the problems that signed distance functions encounter, such as topological artifacts caused due to unreliable normal estimations, which are inevitable given noisy and non-uniform point clouds.
- **[DS]**
  The key insight is this volumetric approach, which computes a crust around the samples point and hierarchicallyrefines the voxel grid with each iteration for efficiency. The crust is computed with a flooding algorithm and then an unsigned distance function is then computed on the crust by dilating from voxels with known function values.
- **[FP]**
  Reconstruction can be formulated and efficiently solved as a graph min-cut problem on a hierarchical voxel structure wrapping the implicit surface.
- **[JD]**
  The key insight is the use of an unsigned distance function instead of a signed distance function. This allows them to process points without any normal information.
- **[LF]**
  The key insight of the paper is a method for calculating an unsigned distance function from only point cloud data. More broadly, the main insight of the paper is to apply graph cuts methods commonly used for image segmentation and 3D stereo reconstruction problems to the point cloud reconstruction problem.
- **[MK]**
  The key idea of the paper is that once a hull containing the reconstructed surface can be computed (e.g. through dilation around the points) then the problem of extracting the surface within the hull can be formulated as a min-cut problem on a graph that has the interior/exterior components of the hull as its source/sink.

3. What are the limitations of the method? Assumptions on the input? Lack of robustness? Demonstrated on practical data?

- **[AS]**
  One limitation of this algorithm is that a min-cut of a graph cannot reproduce every shape. Another limitation is that the octahedral graph generation results in a high memory overhead at higher levels of refinement. Also, the use of voxel grids leads to staircase artifact, which converge slowing to a smooth surface during the smoothing process.
- **[DS]**
  There is significant memory overhead for this method, so the resolution for output models is restricted to 1024^3.
  Because a voxel grid is used, reconstructions of flat surfaces with a slight slope might have staircase artifacts, and they will only slowly converge to a smooth surface.
- **[FP]**
  The dilation step required to set the interior voxels of the surface and adding discarded samples in each refinement level seems to be inefficient in terms of memory allocation (may fill large regions with unimportant voxels) , especially if there are noise or outliers in the sample set.

Another limitation identified by the authors regard the effect that the voxelized structure produce in the triangle mesh. For instance, planes with slight slope will be reconstructed as staircases. This may be overcome smoothing the triangle mesh.

- **[JD]**
  The method is limited to 1024 voxels in each dimension because of memory overhead. The reconstruction can produce stair-stepping due to the voxel grid. Flat areas that are not axis-aligned easily show this.
- **[LF]**
  A large memory overhead as encountered when using the method, in its current form, to obtain output at resolutions greater than 1024^3. The authors say this overhead can be reduced by computing graph cuts results directly on the voxel grid rather than use a graph cuts library that generates another spatial graph structure. The input simply needed to be a set of points in space and no orientation data was required. The method was applied to members of the Stanford 3D model set.
- **[MK]**
  There are no assumptions on the input (points don't need normals).The method is likely not to be robust in the presence of outliers as the method builds up a pseudo-distance function (i.e. using l-infinity distances) so that individual outliers can contribute significantly to the reconstruction.

4. Are there any guarantees on the output? (Is it manifold? does it have boundaries?)

- **[AS]**
  This method guarantees that the output is watertight, of the lowest possible genus, and that it is manifold (although they don't prove this).
- **[DS]**
  The output is a low-genus watertight triangle mesh.
- **[FP]**
  The authors claim that final reconstruction is watertight and manifold. After the segmentation at the finest level, the method constructs a polygonal mesh with non-planar faces. This polygonal mesh is then transformed to a triangle mesh.
- **[JD]**
  The output is a manifold mesh with low genus and a minimum number of vertices.
- **[LF]**
  The output is guaranteed to be watertight. A manifold result can be obtained using post-processing methods. The paper additionally presents a new method for obtaining such a manifold result.
- **[MK]**
  The (primal) method is guaranteed to be watertight because a min-cut on a graph with positive edge-weights has the property that any cycle has to have an even number of cuts. It was not clear if/why the surface was manifold.

5. What is the space/time complexity of the approach?

- **[AS]**
  Minimum cut is the most expensive step in this algorithm. Therefore, the time complexity of this algorithm is O(n^2 log n), where n is the number of points in the point cloud. The space complexity of this algorithm is O(nl), where l is the level of refinement of the voxel grid.
- **[DS]**
  The space is limited by the size of the voxel grid which would depend on the number of iterations/refinements the user specifies.

The time complexity will most likely be limited by the graph cut computation, which can be implemented in O(VE^2) [for a variation of Ford Fulkerson].

- **[FP]**
  The dilation process on each hierarchical level is linear on the number of initial voxels. The graph partitioning step using a min cut – max flow algorithm is O(n3) in the worst case (n is the number of voxels in the level). The construction of the polygonal mesh and the subsequent triangular mesh is be constant for each 2x2x2 block of voxels , therefore it is O(n) on the number of surface voxels. The total memory storage seems to be in the worst case O(h3) where h3 is the resolution in the finest resolution.
- **[JD]**
  The paper does not discuss the complexity but their reference to the min-cut algorithm in BK04 implies an O(EV^2) complexity.
- **[LF]**
  The paper provides the running times for applying the robust reconstruction method to various models in the Stanford 3D model set and obtaining output at varying resolutions (no big oh time complexity was specified). The output resolution of these results is restricted to 1024^3 and the authors do so to avoid a memory overhead arising from use of a certain graph cut library that explicitly generates a spatial graph structure.
- **[MK]**
  Using the octree formulation, the running time should be linear in the number of voxels at the finest resolution that intersect the surface.

6. How could the approach be generalized?

- **[AS]**
  This approach could potentially be generalized to reconstruct higher resolution surfaces by computing the minimum cut directly on the voxel grid, and also visually improved results by incorporating direct solvers.
- **[DS]**
  The resolution of the output can be extended (max is 1024^3) by computing the graph cut directly on the voxel grid.
- **[FP]**
  The space partition can be extended to structures different to voxels. Voxels allow an easy partition and an ordered data structure, but some other type of structures such as tetrahedra (for instance dividing each voxel in 6 tetrahedra) may be considered. The edge weighting strategy and the final construction of a triangle mesh from the “surface voxels” are two fundamental steps susceptible to be interchanged with different approaches than the described in the paper
- **[JD]**
  The method can be applied to any graph that has an unsigned distance function between nodes.
- **[LF]**
  The graph cuts methods used in this paper are already widely used for image segmentation and 3D stereo image reconstruction. The method generates a triangle mesh which can be post-processed at the descretion of the user to obtain a smooth, manifold result.
- **[MK]**
  The approach could be extended to use an adaptive octree, so that at finer resolutions, the problem is only solved on the subset of points that are sampled finely enough. (That way the algorithm would be linear in the input, not the output.)

The approach can be modified to start with constrained Delaunay triangulation of the crust, so that the min-cut would be performed on a graph whose nodes are volumetric cells and hence the surface would be automatically defined as the boundary triangles (rather than requiring the more complex triangulation based on a partition of voxels faces defined by the cuts on the lattice of octahedra).

7. If you could ask the authors a question (e.g. “can you clarify” or “have you considered”) about the work, what would it be?

- **[AS]**
  What do you do in the case where there are two loops of split edges in a voxel (opposite faces)? Is it possible that minimum cut could result in such loops?
- **[DS]**
  Why was the 6-neighborhood chosen for the method, and how would the results be affected if a different neighborhood were used instead (besides potential efficiency / memory issues)?
- **[FP]**
  In the paper the authors defend the position that the confidence function is comparable to the unsigned distance to the surface. Why is this true? Does the confidence function converge to a monotonic increasing sequence from the surface voxels to the crust boundary?
  The selection of the confidence function was experimentally satisfactory but it was not clear why it is theoretically indeed a good choice. What other confidence function would be valid to use?
- **[JD]**
  I would ask the authors about using a smoothing kernel that accounts for the stair-stepping pattern. The error from stair-stepping is regularized with the axes so it may be possible to use an isotropic kernel for smoothing in only those directions.
- **[LF]**
- **[MK]**
  I am unclear as to why the method is guaranteed to generate a manifold output and, hence, why the dual triangulation algorithm described is well-defined.