Spectral Surface Reconstruction from Noisy Point Clouds

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 method presents a new approach to the problem of surface reconstruction from a noisy point cloud without requiring normal information. They introduce spectral methods for partitioning graphs into the problem surface reconstruction. Their algorithm robustly labels tetrahedra as *inside* or *outside* tetrahedra, and is able to find the surface defined by the point cloud as the interface between these two types of tetrahedra. The spectral partitioner has a global view of the point set, and hence is able to ignore noisy points in the point cloud and construct a watertight surface.
- **[DS]**
  The paper presents a noise-resistant method fro reconstructing a watertight surface from a point cloud. It uses a spectral graph partitioning of Delaunay tetrahedra to classify them as "in" or "out", from which a surface triangulation can be extracted. The spectral partitioning algorithm makes local decisions based on a global view of the model. The mtThus, the method is resilient to outliers, holes, and undersampled regions. An extension is provided to obtain a manifold surface.
  The novel part of this algorithm is the spectral partitioning and normalized cut algorithms being applied to surface reconstruction.
  This approach can reconstruct models from clouds with a large amount of noise that other methods cannot do.
- **[FP]**
  The authors propose a method for surface reconstruction in unoriented point sets following a segmentation approach. Instead of constructing the surface just from local based decisions, it is defined a connected graph model over a set of space partitioning structures (specifically Delanauy Tetrahedra), providing a global view of the reconstruction problem. The authors present results proving their method is specially robust to noise and outliers in comparison to local methods such as Tight-Cocone.
- **[JD]**
  The algorithm uses uses spectral decomposition on the Delaunay triangulation of noisy points to patch holes in undersampled regions. The paper introduces spectral partitioning and normalized cuts to the literature. The method can produce surfaces which other methods fail to produce. It is noise-resistant because spectral decomposition uses a global view of the data.
- **[LF]**
  The paper's key contribution is to apply spectral partitioning and normal cuts to the surface reconstruction problem. The resulting method makes graph partitioning decisions based on a global rather than regional view of the graph and in turn is more robust against outliers than methods that take a local approach.
- **[MK]**
  The paper proposes an approach for using spectral clustering to label Delaunay tets as either interior to the solid or exterior. While the spectral approach is novel, the idea of defining a surface as the boundary between interior/exterior Delaunay tets goes back (at least) as far as Boissonnat's 1984 work.

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

- **[AS]**
  The key insight of this paper is that since the spectral partitioner makes local decisions based on its global view of the point set, it can effectively identify the triangles that lie at

the interface between the object being reconstructed and the space around it. Hence, it can be used to identify and ignore points that lie away from the desired surface, and only use those points in the reconstruction that lie close to the surface.

- **[DS]**
  The key insight is the eigencrust method, which labels the Delaunay tetrahedra of the point set by partitioning them into two sets (outside/inside). This is done by constructing a modified Laplacian matrix based on the Delaunay triangulation (and Voronoi diagram) and obtaining an eigenvector by solving a generalized eigenvalue problem.
- **[FP]**
  Pole tetrahedra can be reliably classified as interior or exterior . The classification can be done by partitioning a graph model. Based on this result, we proceed to classify ambiguous tetrahedra.
- **[JD]**
  The partition of the spectral decomposition of the tetrahedra graph into inside and outside tetrahedra identifies the triangles which lie on the boundary. This partition chooses tetrahedra most likely to lie on the interface because it is based on a global view of the points.
- **[LF]**
  The key insight of the paper is to approach the surface reconstruction problem as a segmentation problem and so apply spectral partitioning and normal cuts methods to it.
- **[MK]**
  The approach is motivated by the observation that there tend to be two types of Delaunay tets in the DT. Those associated with the pole vertices, which should be easily identifiable as either inside or outside, and those whose associated Voronoi vertices lie near the surface. In a first pass the authors propose to assign the pole Tets by performing spectral analysis on the graph whose edge weights are defined in terms of the angle of intersection of the spheres associated to adjacent Voronoi vertices. In the second pass the authors classify the remaining tets, again using spectral analysis, this time with edge weights between face-adjacent D-test defined in terms of the isotropy of the shared triangle (to encourage the generation of a surface with a high quality triangulation).

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

- **[AS]**
  This method can occasionally create unwanted handles in the reconstruction, and is also unable to reconstruct sharp corners well. This method also requires that global eigenvalues are calculated two times. This is a slow process, and hence slows the algorithm down significantly. In addition, the computation of the Delaunay tetrahedralization requires that the points be in general position.
- **[DS]**
  This method is much slower than similar algorithms but much more robust to noise.
  It sometimes creates unwanted handles.
  It does not reconstruct sharp corners well.
  It produced good results when run on point sets with noise of variance 2l (where l is the length of the grid spacing).
- **[FP]**
  One of the main limitations of the method is the speed. The eigenvector calculation associated to each spectral graph partitioning is computationally expensive. This performance cost may payoff in noisy point sets, or with large amount of outliers. In the

case of accurately sampled surfaces, the result quality is similar to other more efficient (faster) methods.
The method also seems to produce surface of large genus (i.e., some undesired holes). This is may be an intrinsic characteristic of the spectral graph partitioning, which do not guarantee segmentation in connected components (this contrast to graph-cut segmentation, which tend to generate just two connected components).
The authors identify some flaws in the reconstruction of sharp corners. They claim this is a common feature of tetrahedra labeling methods.

- **[JD]**
  The only assumption on the input is that it contains a recognizable surface. The method works well with many outliers.
- **[LF]**
  The method is said to generally be slower than those that take a local approach to reconstruction and not work as well when reconstructing sharp corners (a common difficulty in surface reconstruction). Input simply needed to be a set of points in space and no orientation data was required. The method was tested on some subjects from the Stanford 3D models set and shown to be much more robust against outliers than the Tight Cocone method.
- **[MK]**
  There seem to be no restrictions on the input (no normals required) and it is designed to provide an implementation that is more robust than previous Delaunay based methods in the presence of the noise. As documented by the authors however, there are no explicit guarantees on the correctness (e.g. in terms of the topology) of the output surface.

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

- **[AS]**
- This method guarantees that the reconstructed surface is watertight. It does not guarantee that the output is manifold, but does describe a post-processing step that guarantees this property.
- **[DS]**
  The output is a watertight surface with an extension that produces a manifold.
- **[FP]**
  Since the reconstruction is obtained by gluing together a set of tetrahedra, the final result is guaranteed to be watertight and closed. A priori, the triangle mesh obtained after the segmentation (i.e., those triangles where inside and outside tetrahedra met) is not manifold.
  Postprocessing of the triangle mesh is required to make it manifold.
- **[JD]**
  The output if guaranteed to be manifold if the optional step is taken. The surface is always water-tight.
- **[LF]**
  The output is guaranteed to be watertight. A manifold result can be obtained using post-processing methods / at the discretion of the user.
- **[MK]**
  The output is guaranteed to be water-tight and, with post-processing, can be made manifold.

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

- **[AS]**

Using Qhull, the time complexity of computing the Delaunay tetrahedralization is O(n log n) , where n is the number of points in the point cloud. However, the eigenvalue decomposition in the spectral partitioning part of the algorithm takes O(n^1.5), which is the most expensive step in the algorithm. Therefore the time complexity of the algorithm is O(n^1.5). The space complexity of this algorithm is O(n).

- **[DS]**
  The time complexity is limited by the eigenvector computation, which depends on the size of the modified Laplacian matrix, which is determined by the number of poles used to construct the graph.
  The space complexity is mostly affected by the construction of the graph. The number of vertices is based on the number of poles used.
- **[FP]**
  The initial Delanauy Triangulation is done over all the point set, so it is worst case O(n2). The construction of the graph is linear, since both the number of vertices and edges is linear on the initial number of samples, and they can be easily found from the Delanauy Triangulation. The computation of the eigenvector required for the spectral graph partitioning is done in O($n\sqrt{n}$) using an iterative Lanczos solver. Finally, the postprocessing stage is linear since all vertex and edges of the triangle mesh are processed at most once, each one with constant time.
- **[JD]**
  The complexity is O(n*sqrt(n)).
- **[LF]**
  The method is said to have an O(n*sqrt(n)) running time on undersampled and noisy models. By constrast, many reconstruction methods simply return poor results given similar models.
- **[MK]**
  I believe that it is O(N^2) for the DT and O(N\sqrt{N}) for the computation of the smallest eigenvector. (Though in practice it seems that DT tends to run faster than O(N^2).)

6. How could the approach be generalized?

- **[AS]**
  This approach should be generalizable to higher dimensions since the notion of Delaunay triangulations extends to higher dimensions, and eigenvalues can also be computed in higher dimensions. This method can also be extended by combining it with the Powercrust algorithm to improve reconstruction at sharp corners.
- **[DS]**
  Sharp corners can be better reconstructed by labeling power cells instead of tetrahedra, which increases the complexity of the model.
- **[FP]**
  The method can be adapted for labeling any partitioning of space. They study the specific case of Delanauy tetrahedralization but some other space partitioning (such as power cells, as they claim) may be considered. However, considering certain space partitioning requires a priori knowledge about reliable structures in this partitioning (such as the pole tetrahedra in the case of Delanauy) in order to define accurate edge weight in the graph.
  They partition the graph using a spectral method (specifically, a variant of normalized cut), but may be some other cut methods could also be used to obtain a graph partition.
- **[JD]**

- **[LF]**
  The methods of spectral partitioning and normal cuts are already widely used for image segmentation and parallel spare matrix arithmetic.
- **[MK]**
  Typical question for Delaunay-based methods: If the sampling density can be estimated, can the method be implemented using the weighted Voronoi diagram?

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]**
  How would you show that the reconstructed surface after the post-processing step that constructs manifolds is in fact a manifold surface?
- **[DS]**
  How can the choice of bounding box affect the results?
- **[FP]**
  To define the graph partitioning they use an eigenvector of the smallest generalized eigenvalue of the Laplacian matrix. What about the eigenvectors of other small generalized eigenvalues? Do they provide any valuable information to enhance the partition?
  At defining the positive edges of the graph, Why did they chose $e^{4-4cos\theta}$ when spheres intersect, and chose no weight when they don’t touch? By just moving two tangent spheres an epsilon appart we pass from having an edge of weight 1 to an edge of weight 0, instead of having a continuous change as would be expected.
- **[JD]**
  Since the matrix L is sparse, can you get the space complexity below O(n^2)? Does your algorithm work out-of-core?
- **[LF]**
- **[MK]**
  When defining the Laplacian, why not just set L_{ii} = -\sum L_{ij}. This would ensure that the constant vector is in the kernel. Thus, the Fiedler vector would have to be perpendicular to the constant functions, and hence would provide a “balanced” partition of positive and negative assignments.
  Why don’t the authors use the values of the eigenvector to identify questionable poles (i.e. define a tet as questionable if the associated coefficient of the eigenvector is close to zero.