Tight Cocone : A Water-tight Surface Reconstructor

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 extends an existing approach to the existing problem of data integration, or generating a surface representation given a set of sample points. The existing approach, called COCONE, does not handle undersamping and produces undesirable triangles and holes in the surface where there is undersampling. The extension to this approach, called TIGHT COCONE, repairs and fills the holes in the surface, producing water-tight surfaces.
- **[DS]**
- **[FP]**
  The paper proposes an algorithm to obtain water-tight (i.e., no holes) surface from an approximate mesh (in this case obtained by Cocone) using a marking and peeling algorithm. This approach, in contrast to previous ones, does not introduce new points and is interpolative.
- **[JD]**
  The paper addresses holes that are due to under-sampling in the cocone algorithm. It guarantees a water-tight surface without adding any points.
- **[LF]**
  This method outputs a water-tight surface given an unorganized set of points. Water-tight reconstructions are needed for many applications (e.g. prototyping, medical imaging, etc.), but are not guaranteed by many reconstruction methods. Addressing this problem means providing a solution that works in practical conditions, that is with rather high sampling error, noise, complex surfaces, etc.
- **[MK]**
  The paper proposes an extension of the PowerCrust (or is it Cocone) method. The original paper had shown that, given sufficiently dense sampling (adapted to the reach) it is possible to guarantee a manifold, water-tight, homeomorophic reconstruction. This paper extends the algorithm to provide a water-tight reconstruction even when the sampling bounds are not met.

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

- **[AS]**
  The key insight of this paper is that holes in meshes generated by popular surface reconstruction algorithms can be filled by the mark and peel algorithm to produce water-tight surfaces.
- **[DS]**
- **[FP]**
  Exploit the known "good" structures of an "approximate" mesh to fill the holes at undersampled regions. In this case, this is done by marking and peeling tetrahedra.
- **[JD]**
  The key insight of the paper is to mark and peel tetrahedra around holes using "good" tetrahedra--ones that are well-sampled--to create a watertight intersection of tetrahedra.
- **[LF]**
  The marking and peeling method for filling all holes and attempting to distinguish intended holes from unintended
- **[MK]**
  As in previous approaches, the idea is to generate a well-defined solid by marking tets of

the Delaunay triangulation as either strictly inside or ourside. A water-tight mesh is then obtained by using the triangles bounding inside/outside tets. The marking is performed using a flood-fill from the boundary, using two criteria:
Where the points are well reconstructed (i.e. the umbrella is well-defined) the triangles of the reconstruction define a natural inside/outside partition and the flood-fill should not be allowed to pass through them.
For bad tets (all vertices not having an umbrella) the flood-fill should only be allowed to pass through faces that are not the smallest.

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

- **[AS]**
  This method relies of the locality of undersampling. If this property is not met, the output surface, although still water-tight, might be improper, for instance, non-manifold or empty, as in the case in the presence of noise beyond the method's tolerance level. Also, the method presented in the paper does not reconstruct internal void.
- **[DS]**
- **[FP]**
  It assumes undersampling is local. In case of noisy inputs it fails to reconstruct a surface. It only reconstructs closed surfaces (i.e., without boundaries).
- **[JD]**
  The algorithm depends on the locality of undersampling. When the undersampling is not local the algorithm fails to generate the right surface. Noisy data can produce this effect as well. The algorithm does not reproduce internal voids because it fills the surface with tetrahedra.
- **[LF]**
  The method assumes that any undersampling will be local. If this assumption is violated the first stage of the algorithm which generates a preliminary surface with the original Cocone algorithm will not be a sufficient representation of the intended surface.
  The method appears to be robust to local undersampling as intended, but is a bit sensitive to noise.
  The method was demonstrated on some difficult datasets but not any that I immediately recognize like the Stanford 3D model set.
- **[MK]**
  The method makes assumes a “principal of locality” to ensure that the second criterion is good, though it never formally defines what this principal is, or why it guarantees anything.

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

- **[AS]**
  When the conditions for which this method is designed (e.g., undersampling, negligible noise) are met, this method guarantees that the output is water-tight, without introducing any extra points in the point sample. It also guarantees that the output is always a subcomplex of the Delaunay Triangulation of the sample points.
- **[DS]**
- **[FP]**
  Output is always manifold. It does not have border. Surface corresponds to exposed faces of interior tetrahedra.
- **[JD]**

The output is guaranteed to be water-tight because it is formed from an intersection of tetrahedra and it is manifold because the cocone algorithm produces a manifold. The output is also a subcomplex of the Delaunay triangulation.

- **[LF]**
  The output surface is guaranteed to be water-tight whether or not the assumptions on input have been satisfied. The output surface may be non-manifold or empty if these assumptions are violated.
- **[MK]**
  The method is guaranteed to generate a water-tight output, though it is not clear that the output is guaranteed to have either manifold edges or manifold vertices.
  Because the method carves out from the convex hull, the initial implementation will only reconstruct surfaces that separate the interior volume from a point at infinity, not the surfaces inside voids. However, the authors propose to fix this by re-running the algorithm from the computed surface, effectively peelings the surfaces off, layer by layer.

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

- **[AS]**
  Both the time and space complexity of this approach is O(n2) in the worst case, since the complexities of this approach are dominated by those of the 3D Voronoi diagram computation. In practice, however, the time complexity is better than quadratic.
- **[DS]**
- **[FP]**
  The initial mesh is computed by Cocone which requires building a Voronoi Diagram of the point set (O(n^2) in the worst case). The marking and peeling phases are linear in the number of tetrahedra of the Delaunay triangulation, therefore they are O(n). Storage is also linear.
- **[JD]**
  O(n^2) in the worst case. They claim it performs better in practice.
- **[LF]**
  Worst case time and space complexity = O(n^2)
  Sub-quadratic in practice
- **[MK]**
  The algorithm requires the computation of a Delaunay triangulation and hence runs, theoretically, in O(n^2) time. However, as observed by Amenta et al., the running time tends to be faster in practice.

6. How could the approach be generalized?

- **[AS]**
  It is not clear that this method can be generalized to higher dimensions. It can, however, be generalized to handle internal voids by modifying the marking and peeling routines slightly.
- **[DS]**
- **[FP]**
  The authors propose an extension of the algorithm to deal with internal voids. They let open the question of how to extend the water-tight reconstruction to a surface with boundary (i.e., how to distinguish between boundaries and holes).
  The Delaunay triangulation, marking and peeling phases are still well defined in larger dimensions (I think so), so the algorithm should be valid for larger dimensions.
  The marking and peeling phases could be adapted to other “approximate” meshes obtained from algorithms different than Cocone (i.e., using a tetrahedral distinct to

Delaunay’s). The initial Delanauy “tetrahedralization” was used to guide the posterior marking and peeling phases, but these phases may still work for similar initial “tetrahedralizations”.

- **[JD]**
  The method uses high confidence values and the principle of locality to fill holes where the confidence is low.
- **[LF]**
  I do not think I understand this area well enough to make a suggestion. Two areas of extension for the method seem to be (1) handling of internal voids and perhaps (2) better distinguishing intended boundaries/holes in a surface from artifacts.
- **[MK]**
  Not clear. Without a formal definition of the principal of locality, it is hard to assess when the method will break down, or what can be done to improve it.

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]**
- **[DS]**
- **[FP]**
- **[JD]**
- **[LF]**
- **[MK]**