ACM Symposium on Solid Modeling and Applications (2004)
G. Elber, N. Patrikalakis, P. Brunet (Editors)

Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies

M. Mortara[†] G. Patanè[†] M. Spagnuolo[†] B. Falcidieno[†] J. Rossignac[‡]

Abstract

Plumber *is a specialized shape classification method for detecting tubular features of 3D objects represented by a triangle mesh. The* Plumber *algorithm segments a surface into connected components that are either body parts or elongated features, that is, handle-like and protrusion-like features, together with their concave counterparts, i.e. narrow tunnels and wells. The segmentation can be done at single or multi-scale, and produces a shape graph which codes how the tubular components are attached to the main body parts. Moreover, each tubular feature is represented by its skeletal line and an average cross-section radius.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Curve, surface, solid, and object representations

1. Introduction

Given a two-manifold closed surface represented by a triangle mesh, *Plumber* automatically extracts the features that can be described as generalized cylinders or cones; we call these features, together with their concave counterparts, i.e. narrow tunnels and wells, *tubular features*. The *Plumber* approach classifies the vertices of a given triangle mesh according to their curvature and shape behaviour in neighbourhoods of increasing size (see Figure 1, 2). Seed vertices are located on tubular features, and clustered to form candidate seed regions which are then used to compute the first reliable tube section, called the *medial loop*, which is ensured to be around each candidate tube and which works as a generator of the feature. Then, the medial loop is moved in both directions on the surface, by using spheres placed at the barycentres of the new medial loops, until the tube is completely swept. The size of the tube is related to the radius of the sphere, and the stop criterion is given by the abrupt variation of the medial loops' lenght. The tube detection is devised in order to work in a multi-scale fashion, where small tubes are detected at first and larger ones at following steps. After the surface segmentation, a geometric representation of each tubular feature is constructed by computing its skeletal line. The configuration of each feature, whose section and length can arbitrarily vary, and its attachments to the body are hierarchically coded in a shape graph.

Different application fields make the surface segmentation an important task. For instance, while tubular structures can be quite easily defined during design processes their automatic extraction from 3D meshes is not a trivial task. We believe that a variety of applications, especially shape recognition and analysis, will benefit if tubular features are identified and abstracted to a centreline and a set of sections. These abstract models, may facilitate the measurements of changes over time in medical applications (e.g. calcification process), or detect abnormalities such as unnatural narrowing or ballooning. Finally, reliable cylindrical models are essential for proper design of prosthetic tubular structures.

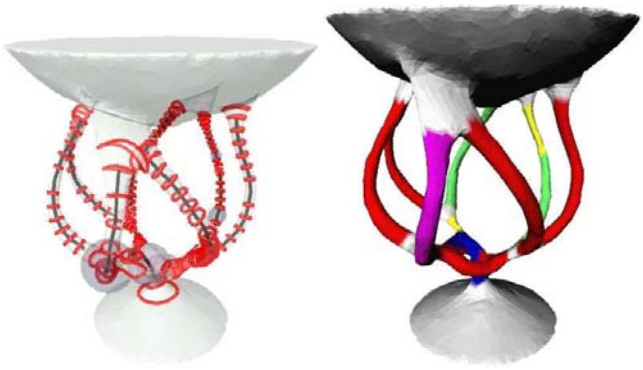

Figure 1: *Tubular features recognized by* Plumber *on a complex model: (a) tube axis and loops, (b) tubes colored with respect to their scale.*

The basic idea of *Plumber* consists of describing a shape by using both local point-wise, and global region-wise measures for shape decomposition and skeleton extraction; in the following, we review the state of the art on those concepts used in the paper.

[†] Istituto di Matematica Applicata e Tecnologie Informatiche, Consiglio Nazionale delle Ricerche

[‡] College of Computing, Georgia Institute of Technology, Atlanta

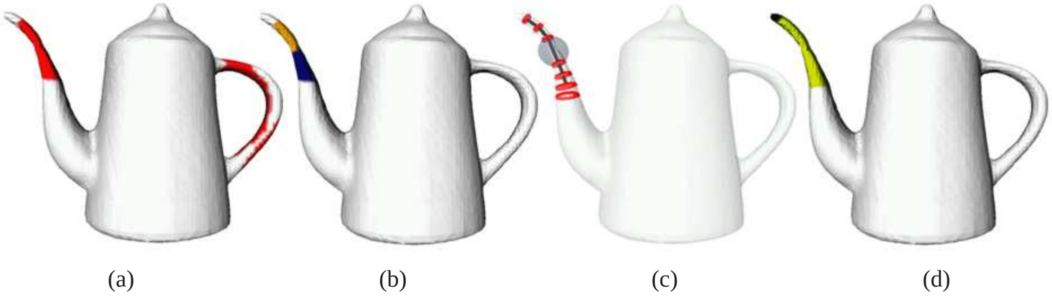

Figure 2: Plumber *method: (a) identification of limb vertices, (b) extraction of their connected components and medial loop, (c) iteration, (d) tube and a cap (black) found at this scale.*

Decomposition methods based on the analysis of the shape boundary evaluate local characteristics to identify patches of the surface that group vertices with similar properties with respect to some measure. In most cases, surface vertices are clustered using the Gaussian curvature: for instance, the segmentation of a free form surface into patches of similar curvature is one of the key steps in reverse engineering [SJTH99, VMC97], and for the validation and verification of visualization products to control mesh quality [ZP01].

In [KT03], the segmentation method is defined as a fuzzy clustering of vertices where the probability that a face belongs to a patch depends on its distance from the other faces of the patch. The advantage of the method is the avoidance of over-segmentation, and that boundaries between adjacent regions are not jagged. The results show that the segmentation is meaningful, in the sense that the extracted components locate the main natural features of the objects.

Skeletons such as the Medial Axis Transformation (MAT) and the Reeb Graph assume that the surface is the boundary of a volume, and analyse the shape according to its interior, thus providing descriptions which better highlight its global structure.

The MAT is constructed using the paradigm of the maximal enclosed spheres, whose centres define a locus of points which describes, together with the associated radius, the width variation of the shape. The MAT of a 3D surface is generally a non-manifold complex, computationally heavy, and sensitive to noise because tiny perturbations may produce a whole new arc. Furthermore, there is not a direct relation between tubular features and specific components of the MAT, especially when the tubes have an arbitrary shape and the cross sections do not exhibit any symmetry.

More relevant for the identification of tubular features are methods for the extraction of skeletons, which provide an abstract shape representation by a graph of lines that retain the connectivity of the original shape. The Reeb graph [SKBT96, VL00, BMMP03] is a topological structure which codes a given surface by storing the evolution of the level sets of a mapping function defined on its boundary.

In [LTH01], tubular parts are identified using a sweeping techniques along the arcs of the skeleton which is constructed by joining the edges remaining after an edge collapse process on the whole mesh. These edges are linked in a tree structure, and it is used as a support for the sweeping process where the mesh is intersected by a set of planes and tubes are identified by looking at the geometry of the cross-sections.

The main difference between *Plumber* and segmentation methods previously discussed is that we extract building primitives of the object with a specific structure, i.e. generalized cones and cylinders, and not only related to a curvature and concavity analysis [MPS*04, KT03]. Furthermore, while skeletal representations do not provide a scale-based decomposition of the shape and are usually unstable with respect to wripples or wrinkles, *Plumber* differentiates tubular features of different dimension.

The reminder of the paper is organized as follows; in section 2 the *Plumber* method is detailed, and the discussion on possible applications concludes the paper.

2. The *Plumber* method

Intuitively, ideal tubes are identified by parts of the shape whose intersection with a sphere of appropriate radius produces two intersection curves. The section of the tube and its axis can be arbitrarily shaped, and the size of the tube is kept as a constraint during the identification process. Chosen a level of detail R, *Plumber* performs the following steps:

1. identify *limb-regions* associated with at least two loops on M (see Figure 2(a));
2. shrink each of the two selected boundary components along the surface to its medial-loop, whose points are nearly equidistant from the two border loops (see Figure 2(b));
3. expand-back the medial-loop by sweeping the extent of the shape in both directions. More precisely, at each iteration we place a sphere of radius R in the barycentre of the new medial loops. If the intersection between the sphere and the surface generates two loops, mesh vertices inside the sphere are marked as visited;
4. the procedure is iterated in both directions until:
   - no more loops are found, or more than one loop is found on not-visited regions;
   - the new loop lies on triangles that are already part of another tube, or the length of the new loop exceeds a pre-defined threshold.
5. the tube skeleton is extracted by joining the loops' barycentres.

The previous steps are detailed in the following paragraphs.

Vertex classification Given an increasing set of radii R_i, $i = 1, \ldots, n$, *Plumber* characterizes a 3D mesh M in a neighbourhood of a vertex p at the scale R_i by analysing the evolution of the connected components of the curve $\gamma_i := M \cap S(p, R_i)$, where $S(p, R_i)$ is the sphere of center p and radius R_i [MPS*04, MP02]. The following classification is used:

- *1 boundary*: the surface around p is considered topologically equivalent to a disc (see Figure 3(a)).
- *2 boundary components*: the surface around p is tubular-shaped (see Figure 3(b)). Their lengths are used to distinguish between conic and cylindrical shapes, and p is classified as a *limb-vertex*.
- *$n \geq 3$ boundary components*: in a neighbourhood of p a branching of the surface occurs (see Figure 3(c)).

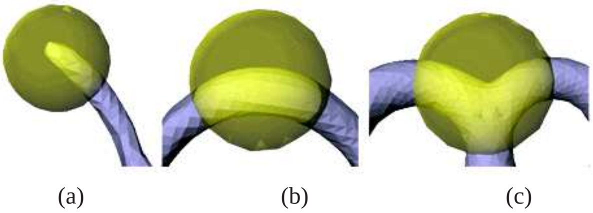

Figure 3: *Different cases of sphere to surface intersection.*

Intersecting the mesh with a sphere with radius R_i allows to identify limb-vertices if they lay on a tube of diameter R_i or smaller. At each vertex $p \in M$, we consider three spheres of radius $R_i - \varepsilon$, R_i, and $R_i + \varepsilon$ with ε given threshold proportional to the minimum edge in the triangulation. We consider limb vertices those ones whose curve γ_i has two or more boundary components (see Figure 4). This classification improves stability for identifying tubes of arbitrary cross section where isolated limb-vertices could appear; a stricter choice consists of considering as limb vertices those ones classified with the same label at all the three scales.

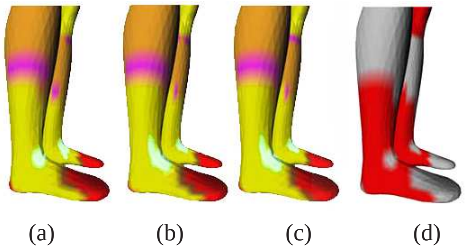

Figure 4: *In yellow limb-vertices found at scale $R - \varepsilon$ (a), R (b), and $R + \varepsilon$ (c). All the limb vertices are depicted in (d).*

The choice of the set $\{R_i\}_{i=1,\ldots,n}$ is related to the scale of the features which have to be extracted, and for performing a multi-scale analysis of the shape; small radii determine details, while bigger ones are used to analyse the global characteristics of the surface. Further discussions are given in the paragraph *Mutli-scale analysis*.

Identification of tube candidates from limb vertices The second step is the identification of the maximal edge-connected components of limb-vertices, using a depth-first search. Note that while the analysis of the evolution of γ_i produces a vertex-oriented classification of M, regions composed by limb-vertices are not guaranteed to be tube-shaped as a whole (see Figure 2(a), on the handle). In particular, limb regions may have not two boundary curves. Therefore, the next step defines a criterion for judging if a limb-region is a good candidate for the tube identification; that is, the limb-region is *around* the tube. For instance, in the case that the tube section is ellipsoidal and its size is greater than the chosen scale, it may happen that the spheres used to classify the vertices produce only one intersection curve on one side of the tube, and two on the other side (see Figure 5), thus giving rise to a limb-region not surrounding the tube. Therefore, the region is not tube-shaped at the given scale and it has to be discarded; it will be found at a larger scale.

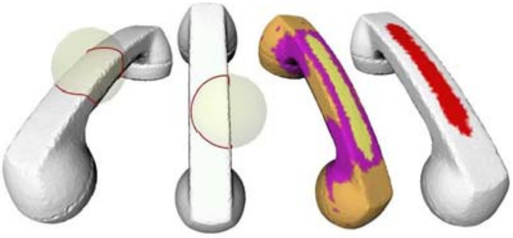

Figure 5: *Example of limb-regions (in yellow) whose vertices on γ have one boundary component.*

Medial loop generation Seed tubular regions are used to construct a medial loop around each candidate tube that will be used for the tube identification and its centreline construction.

Because we have already detected all the candidate tubular regions, a seed point for each tube is selected; for instance, we could choose the centroid of each region, i.e. the point with maximum distance from the region boundary, and then generate the loop with one of the methods proposed in [VF02, GW, LPVV01]. Instead, *Plumber* relies on the limb-region boundaries which are loops surrounding the tube. The idea is to find the medial loop by moving the boundary loops in the middle of the limb-region; to this end, we perform a morphological shrink by simultaneously invading the component from its two boundary components.

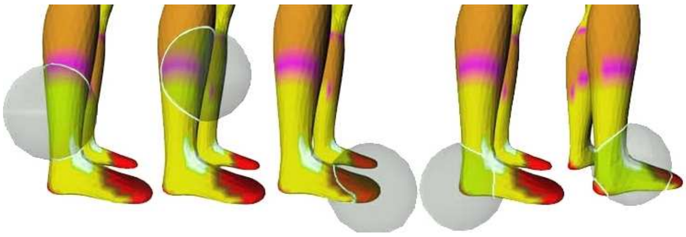

Figure 6: *A tubular region affected by small features, like the neel. The configuration of the sphere/mesh intersection is depicted, with spheres centred in vertices of different feature types: (a) limb, (b) blend, (c), (d) tip, (e) split.*

Firstly, the two boundary components of the limb-region (the two of greater length if the region has three or more border loops, as in Figure 6(e)) are computed. Let R and L (for "right" and "left" respectively) be the two boundary components of the tube; at first, each vertex p on L is associated with the couple $(0, +\infty)$ that indicates that p has distance 0 (resp. $+\infty$) from the boundary L (resp. R). The same initialization applies to R.

The distance of a vertex p from one boundary is computed as the shortest edge path connecting p with a boundary vertex. Then, the distance values of all points are updated, propagating from L and R towards the interior of the region. The distance propagation from L will update the first value of the distance vector, while that from L will affect the second field; at the end of the process, vertices are classified as nearer to L or R (see Figure 7). Edges connecting vertices of different type are cut by the medial loop we are looking for; that is, we join the mid-points of those "medial edges" to produce a medial loop. This construction achieves two good effects with respect to other methods [Kar99, VF02]: the medial loop is guaranteed to be non-trivial and inside the region. The non-minimality of its length does not affect the growth of the tube, and the construction of the skeleton. In the case of three or more boundary components, the choice of starting from the two loops of greater length is to guarantee a stronger reliability to the tube extraction with respect to smaller intersection curves which may be due to local undulations of the shape.

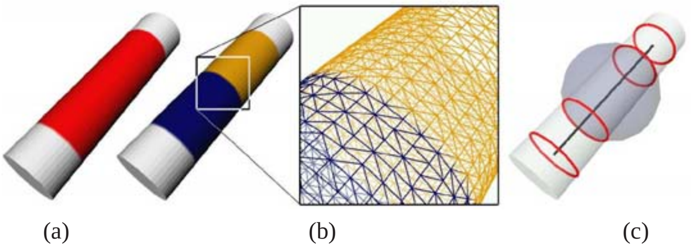

Figure 7: *(a) Limb vertices, (b) connected component of the limb vertices with two boundary components, and medial loop (marked curve), (c) medial sphere centred in the barycentre of the medial loop, and tube growing.*

Loop expansion and skeleton construction The loop expansion is controlled by a verification procedure which, at each step, extends the center-line and at the same time ensures that the surface is tubular around it. A first medial sphere is drawn, whose centre p is the barycentre of the medial loop, and whose radius is R. If $M \cap S(p,R)$ has not two boundary components, the growing stops and the candidate tube is discarded. Otherwise, a new sphere with the same radius is centred in the barycentre of the two loops; the process is then split into two parts, trying to grow the tube in both directions. Now we focus on the sphere moving in one of the two directions, since the other case is symmetric.

At each iteration, the sphere rolls to the barycentre of the next loop, and the triangles laying completely or partially inside the sphere are marked as belonging to that tube. Then, the intersection between the sphere in the new position and the mesh is again computed, taking into account only the intersection curves through non visited triangles (all the spheres except the medial one have always a "backward" loop, passing on the already marked triangles). During the loop expansion, the following cases may arise:

- no intersection curves are found. This is the case of a tubular protrusion terminating in a tip; visited triangles locate a *cap* (see Figure 8(a), in the square);
- the intersection curve consists of one loop (see Figure 8(a)). If its length is less than a pre-defined threshold, the size of the tube section is not varying too much; the loop becomes a new cross section and its barycentre contributes to the skeleton as a new node. Otherwise (see Figure 8(b), in the oval), the growth stops.
- the intersection counts two, or more loops; that is, a bifurcation occurs (see Figure 8(b)). The growing of the tube in this direction stops, and the last visited triangles are unmarked.

Finally, the barycentres of the medial loops are joined to define the tube skeleton.

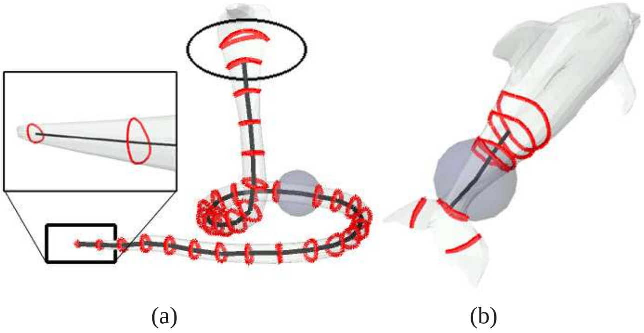

Figure 8: *(a) No new loop is found on the snake tail (in the box), and a loop discarded after the length check on the head (in the oval). (b) A branching occurs on the dolphin tail.*

Multi-scale analysis The extraction of tubes at scales $R_1, \ldots, R_n$ adopts a fine-to coarse strategy, marking triangles as visited while the tube grows and which are not taken into account during the following steps (see Figure 9). Analogously, the medial loop computation simply does not take into account smaller tube vertices, propagating distance values only on not-visited vertices (Figure 9(d)). Decisions are taken when the loop passes partially on not-visited and tube triangles. For example, in the case depicted in Figure 9(e), the two smaller loops fall on tube triangles, and are not counted; therefore, this is the case of two intersection loops, and not that of a branching. The tube is grown, and the result of the two iteration steps is shown in Figure 9(f). This set of radii is selected by the user, or assigned by uniformly sampling the interval from the minimum edge lenght in M to that of the diagonal of its bounding box.

At the end of the whole process, tubes are labelled with respect to the scale at which they were found. The connected components of the shape parts which are not classified as tubes or caps define *body* parts of the object, and the resulting decomposition is coded in a tube-body connectivity graph which represents the spatial arrangement of the tubular features onto bodies. A smooth transition of radii ensures a meaningful growth of the tube at a scale R_i, while discarding smaller features and analysed at the previous levels of detail $R_j, j = 1, \ldots, i-1$.

Strict/ non strict mode Together with the size of the sphere rolling over the centreline, the other parameter to be fixed in the tube growing step is the threshold in the loop length check. To this end, we stop the growing when the tube becomes too large, i.e. the length of the intersection loops varies too much.

If we consider a natural object, we probably do not want to decompose natural limbs into pieces; on the other hand, in the case of a manufactured model, we may want to be precise with respect to the tube size, eventually splitting a tube into components of

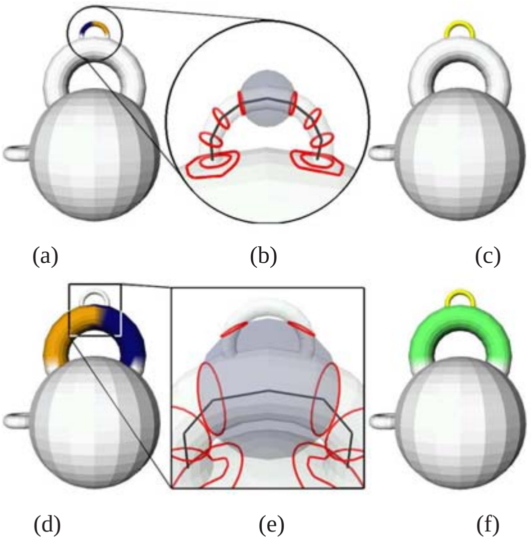

Figure 9: *Iteration of* Plumber *at increasing scales.*

(even slightly) different sections. For this reason, two alternatives are available (see Figure 10):

- a *strict* mode, useful in CAD and medical applications,
- a *non-strict* mode, for other applications where a continuous variation of the tube size does not require to split the tube.

In the strict mode, each time a new loop is generated its length is compared with that of the intersection loops associated to the medial sphere at the beginning of the process, and not with the length of the medial loop tube section which can be non minimal and misleading. Other choices were also taken into account, such as average, minimum, and maximum tube length; the one adopted is a compromise between the required strictness and robustness. In the non strict mode, a loop is accepted if its length is less or equal to twice the length of the previous loop. In both cases, the user can select values on the base of a-priori information or specific needs.

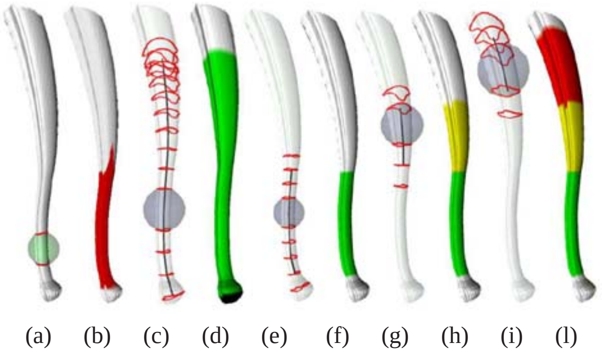

Figure 10: *(a) Initial level of detail, (b) limb-region, (c) tube growing in non-strict mode, and (d) tube extraction. In (e) tube growing from the same limb-vertices in strict mode, (f) the extracted tube, (g), (h) (i) next iterations, (l) achieved segmentation at the chosen scale.*

Table 1: Plumber *timings (m:s:ms) performed on an Athlon 1000 MHZ.*

Model	n_V	Vert. classif.	Medial loop	Tube grow
Cylinder	4038	00.26.84	00.15.45	$< 1sec.$
Pot (1 iter.)	14616	00.08.47	00.03.24	$< 1sec.$
Pot (2 iter.)	–	00.13.45	00.13.50	$< 1sec.$
Schale	10892	00.19.65	02.25.96	$< 1sec.$

Shape graph Throughout the previous paragraphs we have detailed a method for identifying and classifying tubes of different size and bodies achieving a segmentation of the input object. We enrich this geometric classification with an explicit representation of the structure of the model which codes the relations between primitives in a hierarchy of tubes and bodies. This structured representation is a *shape-graph* whose nodes are the extracted primitive shapes, while the arcs code the adjacency relation among the previous ones, i.e. their relative position and orientation. Each node is a *tube*, whose labels are the medium radius and the axis length, a *body*, whose labels are the number of holes and the approximate volume, or a *cap*, whose labels are the basis section, the axis length and the curvature extrema. Each arc between two adjacent nodes falls into one of these cases (see Figure 11): tube-body, tube-tube, cap-tube. The tube-body or tube-tube adjacency is called H-junction (i.e. *handle-junction*) if both boundaries of the tube lay on the same body or tube respectively; in this case, the arc is a loop and the tube locates a handle on the input object. In the case that only one boundary of the tube belongs to the tube-body the adjacency is called a T-junction.

Computational complexity The predominant cost of the method is represented by the initial surface characterization [MPS*04] to detect limb vertices, which is $O(n_V^2)$ with n_V number of mesh vertices. The following tube extraction procedure is much faster. The clustering of limb vertices into regions is treated triangle-wise; starting from a first seed triangle having three limb vertices, the region is constructed adding neighbouring limb triangles through a breadth-first search. The boundary computation of a region is linear in the number of vertices of the region: all the vertices are visited, and when a seed boundary vertex is found, the boundary loop which it belongs to is computed moving by adjacency.

The medial loop computation is in the worst case very expensive: the problem of computing the minimal distance between two vertices can be solved by the Dijkstra's algorithm in $O(nlog(n))$, where n is the region cardinality, but in our case the minimum distance from all the boundary points takes $O(n^2log(n))$. In practice boundary vertices are much less than n, about $n^{1/2}$, thus reducing time complexity in the average case. The tube growing procedure consists at each step in a triangle visit, starting from those laying on the previous medial loop, until a triangle intersected by the sphere is found. Each triangle inside the sphere is visited once, and the computation of the intersection curve itself is linear in the number of intersected triangles, determined by adjacency. So each tube is grown in linear time with respect to the number of triangles it includes. Timings are reported in Table 1.

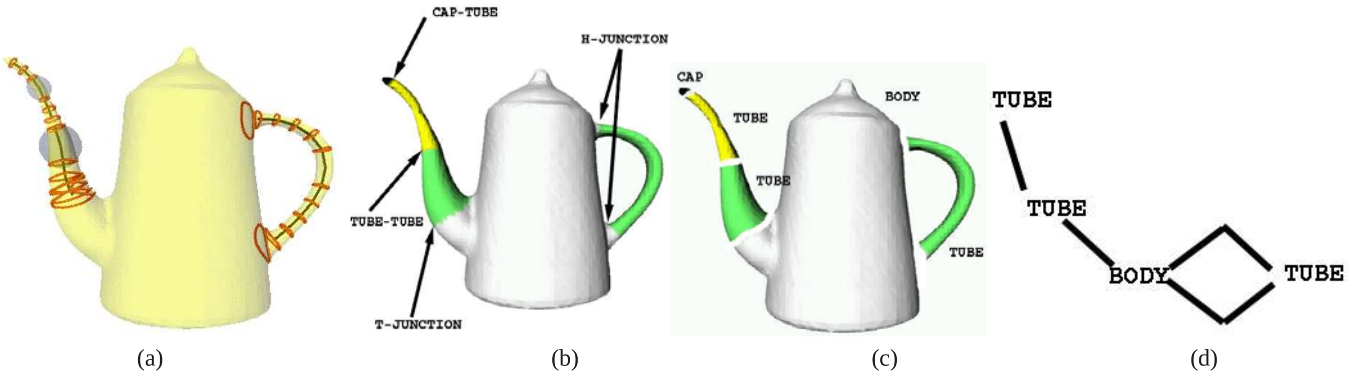

Figure 11: *(a) Centrelines on a tea-pot with respect to two levels of detail, (b), (c) Segmentation of the tea pot into cap, body, tubes and adjacency relations, (d) shape graph.*

3. Applications and Conclusions

The *Plumber* algorithm provides a multi-scale method to decompose a complex shape into its tubular features and bodies. The segmentation considers as bodies those regions that are not tubular shaped; therefore, a sub-classification of these primitives is necessary. Main difficult tasks are their identification, general configuration and the identification of a basic shape for the abstraction.

The interpretation and categorization of tubular features has the drawback of introducing heuristic thresholds to make decisions on the tube size, or for distinguishing branching parts from complicated configurations of tubes as in Figure 1, and 9. The reduction of the influence of these parameters and the abstraction of tubular features with generalized cylinder and cones for collision detection applications are the further improvements of *Plumber*.

4. Acknowledgements

This work has been supported by the Research Agreement *Surface Analysis* between GVU/Gatech and IMATI-GE/CNR, and the national Project “MACROGeo-Algorithmic and Computational Methods for Geometric Object Representation". Thanks are given to the Shape Modelling Group at IMATI-GE/CNR.

References

[BMMP03] Biasotti S., Marini S., Mortara M., Patanè G.: An overview on properties and efficacy of topological graphs in shape modelling. In *Shape Modeling International* (2003), pp. 10–15. 2

[GW] Guskov I., Wood Z.: Topological noise removal. In *Graphic Interface*, pp. 19–26. 3

[Kar99] Kartasheva E.: The algorithm for automatic cutting of three dimensional polyhedrons of h-genus. In *Shape Modeling International* (1999), pp. 10–15. 4

[KT03] Katz S., Tal A.: Hierarchical mesh decomposition using fuzzy clustering and cuts. *Transactions on Graphics 22*, 3 (2003), 954–961. 2

[LPVV01] Lazarus F., Pocchiola M., Vegter G., Verroust A.: Computing a canonical polygonal schema of an orientale triangulated surface. In *Symposium on Computational Geometry* (2001), pp. 10–15. 3

[LTH01] Li X., Toon T. W., Huang Z.: Decomposing polygon meshes for interactive applications. In *Symposium on Interactive 3D graphics* (2001), pp. 35–42. 2

[MP02] Mortara M., Patanè G.: Shape covering for skeleton extraction. *International Journal of Shape Modeling 8*, 2 (2002), 139–158. 3

[MPS*04] Mortara M., Patanè G., Spagnuolo M., Falcidieno B., Rossignac J.: Blowing bubbles for the multi-scale analysis and decomposition of triangle meshes. *Algorithmica, Special Issues on Shape Algorithms 38*, 2 (2004), 227–248. 2, 3, 5

[SJTH99] Sacchi R., J.F. P., Thomas P., Hafele K.: Curvature estimation for segmentation of triangulated surfaces. In *3-D Digital Imaging and Modelling* (1999), pp. 536–544. 2

[SKBT96] Shinagawa Y., Kunii T., Belayev A., Tsukioka T.: Shape modeling and shape analysis based on singularities. *International Journal of Shape Modeling 2*, 1 (1996), 85–102. 2

[VF02] Verroust A., Finiasz M.: A control of smooth deformations with topological change on a polyhedral mesh based on curves and loops. In *Shape Modeling International* (2002), pp. 10–15. 3, 4

[VL00] Verroust A., Lazarus F.: Extracting skeletal curves from 3d scattered data. *The Visual Computer 16*, 1 (2000), 15–25. 2

[VMC97] Várady T., Martin R. R., Cox J.: Reverse engineering of geometric models - an introduction. *Computer-Aided Design 29*, 4 (1997), 255–268. 2

[ZP01] Zhou L., Pang A.: Metrics and visualization tools for surface mesh comparison. In *Symposium on Electroning Imaging: Science and Techmology* (2001). 2