Using the Skeleton for 3D Object Decomposition

Luca Serino, Gabriella Sanniti di Baja, and Carlo Arcelli

Istituto di Cibernetica "E.Caianiello", CNR
Pozzuoli, Naples, Italy
{l.serino,g.sannitidibaja,c.arcelli}@cib.na.cnr.it

Abstract. An object decomposition method is presented, which is guided by a suitable partition of the skeleton. The method is easy to implement, has a limited computational cost and produces results in agreement with human intuition.

1 Introduction

Object decomposition is of interest in the framework of the structural approach to description and recognition to reduce the complexity of the task. In fact, as discussed in [1-4], the human visual system may represent objects with complex shape in terms of simpler parts, by decomposing the objects into parts and by organizing object representation in terms of the parts and of their spatial relationships. The main advantage of such a structured representation is a greater robustness under changes in viewing conditions.

Since the appearance of the above papers, several methods have been proposed for object decomposition. For example, decomposition has been guided by skeleton partition [5-8], distance information [9,10], diffusion distance [11], spectral clustering [12], modal analysis [13], and classification within triangle mesh object representation [14,15].

Decomposition guided by skeleton partition has been followed particularly for objects than can be perceived as articulated in parts with tubular shape, where a one-to-one correspondence exists between the individual curves composing the skeleton and the individual parts of the object. In this case, it is convenient to associate the points of the skeleton with their distance from the complement of the object. Then, the union of the balls centered on the points of a given curve of the skeleton and with radii equal to the associated distances identifies a perceptually significant object part. In turn, when the object also consists of parts that cannot be interpreted as 3D generalized cylinders or cones, individual skeleton curves do not necessarily correspond to perceptually significant object parts. Thus, some clever grouping of skeleton curves is necessary to generate a skeleton partition where each partition component corresponds to a perceptually meaningful object part.

We also note that the partition of the skeleton into its constituting individual curves may result in a decomposition where the separation between adjacent parts does not occur in correspondence of significant curvature changes along the boundary of the object. This problem is due to the fact that the skeleton of an object that is not articulated into generalized cylinders or cones actually represents a rather sketched version of the object where large parts of the original object are not recovered by the union of

A. Heyden and F. Kahl (Eds.): SCIA 2011, LNCS 6688, pp. 447–456, 2011.
© Springer-Verlag Berlin Heidelberg 2011

the balls associated with the skeleton points. As a consequence, in the obtained object decomposition the separation between adjacent parts is in correspondence with the curvature changes along the boundary of the sketched version of the object, rather than along the boundary of the original object.

In [8] we introduced a criterion to decompose a 3D object by suitably partitioning the skeleton. Here, we present an improved method whose strategy is as follows. A partition of the skeleton in three different kinds of subsets is performed by taking into account the notion of *zone of influence*. Then, the components of the partition are used to build object's parts. Since pairs of adjacent parts may be separated by non-planar surfaces, a concavity filling procedure is employed to redistribute voxels in each pair of adjacent parts in such a way to obtain an almost planar separation. Finally, a merging process is accomplished to obtain a decomposition of the object into a smaller number of perceptually significant parts.

The main differences with respect to the work in [8] regard: i) the way in which the partition of the skeleton is performed (two kinds of skeleton partition components were considered before), ii) the introduction of suitable criteria to manage cases in which an individual part may result as split into sub-parts, and iii) the suggestion of a more powerful merging strategy to reduce the number of parts to the most significant ones.

2 Preliminaries

We refer to solid objects, i.e., rid of cavities, in binary voxel images in cubic grids. The object is the set of 1's and the background is the set of 0's. We use the 26-connectedness for the object and the 6-connectedness for the background.

For an object voxel p, the 3×3×3 neighborhood N(p) includes the six face-, the twelve edge- and the eight vertex-neighbors of p.

Given two voxels p and q, their distance is measured as the length of a minimal discrete path linking p to q. We use the weights w_f=3, w_e=4 and w_v=5, as suggested in [16], to measure moves from a voxel towards its face-, edge- and vertex-neighbors along the path, respectively. This choice of weights is motivated by the fact that the <3,4,5> weighted distance provides a reasonable approximation to the Euclidean distance.

The skeleton is a subset of the object consisting of curves symmetrically placed within the object, with the same topology of the object, and such that each point of the skeleton is associated with the value of its distance from the background, i.e., the radius of a ball that, centered on the point, is tangent to the object's boundary and is included in the object. Skeletonization has been influenced by the notion of medial axis transform introduced by Blum [17], and a number of papers dealing with the computation of the skeleton can be found in the literature (e.g., see [18] and the references quoted therein). In this work, we use the skeletonization algorithm suggested in [19] to guide object's decomposition. Skeletonization aims at the inclusion in the skeleton of the centers of maximal balls of the object, i.e., the voxels whose associated balls are included in the object but are not completely included by any other single ball in the object. In fact, inclusion of all centers of maximal balls, CMB, guarantees that the object can be fully recovered by applying to the skeleton the reverse distance transformation [16]. However, for 3D objects the CMB generally concentrate along symmetry planes and axes, so that not all CMB of the object can be kept as skeleton points.

The ball associated to a distance labeled voxel *p* is obtained by applying to *p* the reverse distance transformation. Balls associated to a set of possibly sparse voxels may overlap and merge into connected components. Each group of balls forming a connected component is called zone of influence of the set of distance labeled voxels it includes. Distance labeled voxels that are neighbors of each other or are closer to each other than the sum of the corresponding radii are included in the same zone of influence.

A voxel *p* of the skeleton is an end point when it has only one neighboring skeleton voxel in N(*p*).

A voxel *p* of the skeleton is a branch point when it has more than two neighboring skeleton voxels in N(*p*). The zones of influence obtained by applying the reverse distance transformation to the branch points will be used in this paper.

Concavity filling is a process that identifies voxels of the background placed in local concavities of the object and adds them to the object. Concavity filling can be iterated as far as local concavities are detected. The concavity filling algorithm used in this paper is based on the use of 3×3×3 operations [20].

3 Object Decomposition

Our method provides a one-to-one correspondence between skeleton subsets and object parts. It includes the following tasks: i) skeleton partition into three types of subsets, called *simple curves*, *complex sets*, and *single points*; ii) recovery of object parts associated to simple curves, single points and complex sets, respectively called *simple regions*, *bumps and kernels*, where the latter regions constitute a sort of main bodies of the object, from which simple regions and bumps protrude; iii) making planar the separating surface between adjacent recovered parts; iv) managing the case of parts resulting erroneously split into sub-parts; and v) expanding simple regions, bumps and kernels.

To describe our procedure, let us consider the example shown in Fig.1.

Fig. 1. The object “camel”, left, and its skeleton, right

3.1 Skeleton Partition

We apply the reverse distance transformation to the branch points (see Fig. 2 left). Then, as said in the previous section, a number of connected components, i.e., the zones of influence of the branch points, is obtained which is smaller than or equal to the number of branch points. The zones of influence are used to partition the skeleton into simple curves, complex sets, and single points. A simple curve is a connected component of skeleton voxels that are not included in any zone of influence. As regards skeleton voxels included in zones of influence, we distinguish two cases. When end points are not included in a zone of influence, all skeleton voxels included therein constitute a complex set (see the paws "camel"). In turn, if end points are included in a zone of influence, the end points themselves constitute single point partition components, while the remaining skeleton voxels in the zone of influence constitute a complex set (see the body of "camel"). In Fig. 2 right, the three types of partition components are shown in different colors (green for simple curves, gray for complex sets, and red for single points).

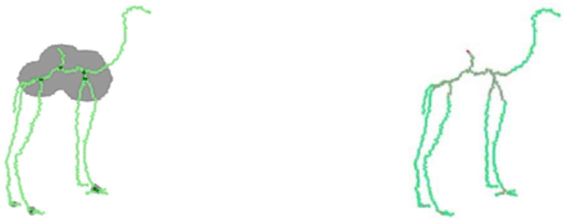

Fig. 2. The zones of influence, shown in gray, obtained by applying the reverse distance transformation to the branch points, left, and the three types of partition components, right

Each partition component is assigned an identity label that accounts for the component type and distinguishes the components of the same type.

3.2 Recovery of Simple Regions, Bumps and Kernels

The reverse distance transformation is applied to the individual partition components, and the identity label ascribed to the partition component is, in principle, assigned to the object voxels recovered by that component. Actually, individually recovered parts partially overlap, so that more than one identity label is possible for some recovered voxels. Thus, to have a decomposition into disjoint parts, we identify the connected components of recovered voxels with more than one identity label and ascribe to the voxels in each of these components the identity label pertaining to the kernel that overlaps the component itself.

Note that for a complex set with end points taken as single points, we do not apply the reverse distance transformation to voxels that, in the skeleton, linked the end points to the remaining skeleton voxels of the complex set. This choice avoids that the surfaces separating bumps from the adjacent kernels are misplaced with respect to the intuitively expected positions. This topic will be treated in more detail in Section 5.

Fig.3 left shows the result of the first recovery step.

Fig. 3. Simple regions, bumps and kernels after the first recovery step, left. Decomposition of "camel" where each skeleton partition component is associated with an individual object part, right.

3.3 Improving the Separation between Parts

The surface separating a bump or a simple region from an adjacent kernel may be not planar. An example is shown in Fig. 4 left, where a simple region of "camel" (the head-neck region) is shown.

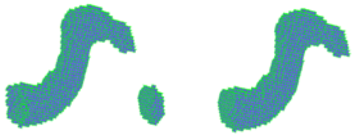

Fig. 4. A simple region before, left, and after, right, concavity filling. Voxels shown in the middle are the voxels removed from the adjacent kernel and added to the simple region to make planar the separating surface.

Concavity filling is employed to assign to a bump or to a simple region the voxels placed in local concavities of that region and belonging to an adjacent kernel. By iterating concavity filling as far as local concavities are detected, the surface separating the region from the adjacent kernel becomes almost planar. Since concavity filling uses 3×3×3 operations that are able to derive curvature information from a 5×5×5 neighborhood [20], the separating surfaces reasonably well approximate planar surfaces.

As an example of the effect of concavity filling see Fig. 4 right.

3.4 Removal of Sub-parts

We note that a part with a given identity label may result as divided into sub-parts, due to i) the criterion described in Section 3.2 to manage overlapping among regions, and ii) to the re-assignment of identity label during concavity filling. We want that only one connected part is associated a given identity label, so that a one-to-one correspondence exists between partition components and object parts. To this purpose, whenever sub-parts with the same identity label are detected, we discriminate them into significant sub-parts and non-significant sub-parts. Only significant sub-parts are preserved, while non-significant sub-parts are set to the background value zero.

A sub-part of a simple region or of a kernel is regarded as significant if it includes at least one skeleton voxel. This criterion cannot be used for bumps, since they definitely do not include skeleton voxels. Thus, if a bump results split into sub-parts, the criterion is to take as significant only the sub-part with the largest size.

3.5 Expanding Simple Regions, Bumps and Kernels

As already pointed out, the original object cannot be completely recovered starting from the skeleton, so that bumps, simple regions and kernels obtained so far are parts of a sketched version of the input object, rather than of the whole original object.

To decompose the input object, expansion of bumps, simple regions and kernels is performed over the voxels of the input object that were not recovered from the skeleton. Object voxels reached by the expansion process are assigned the identity label of the part they are closer to. Voxels at the same distance from more than one part are assigned the label of the part where they have the largest number of neighbors. The resulting decomposition for "camel" is shown in Fig. 3 right.

4 Merging

Though the obtained decomposition satisfies the one-to-one correspondence between skeleton partition components and object parts, the number of parts may be not in accordance with human intuition, so that merging criteria to reduce the number of parts to the perceptually most relevant ones should be devised.

Since we regard kernels as constituting the main bodies of the object, we aim at merging to kernels suitable bumps and simple regions protruding from them.

We distinguish simple regions into peripheral regions, i.e., adjacent to one kernel only, and non-peripheral regions, i.e., delimited by two kernels.

We first candidate to merging only non-peripheral simple regions. To decide whether any such a region should be merged into a unique object part together with the two delimiting kernels, we use the following visibility criterion.

For the current simple region we consider its surface, which consists of the voxels having a face-neighbor in either the background or any of the delimiting kernels. The total area of the surface is given by the number of faces of the surface voxels of the simple region that are 6-adjacent to the complement of the simple region. The surface is interpreted as consisting of two portions: a visible portion and a non-visible portion. The area of the visible portion, A_v, consists of the number of faces that are 6-adjacent to the background. The area of the non-visible portion, A_{nv}, is given by the number of faces 6-adjacent to the delimiting kernels.

If the ratio A_v/A_{nv} is smaller than an a priori fixed threshold θ, the simple region is regarded as scarcely visible and is merged with the adjacent kernels into a unique component. In this paper, θ has been set to 2 by considering a continuous cylinder with height h, for which it is $A_v/A_{nv}= 2\pi rh/2\pi r^2$, as scarcely visible if h is smaller than the diameter $2r$. To appreciate the effect of merging, see Fig. 5, where the object "horse", its skeleton, the decomposition before merging non-peripheral simple regions, and the result after merging are shown.

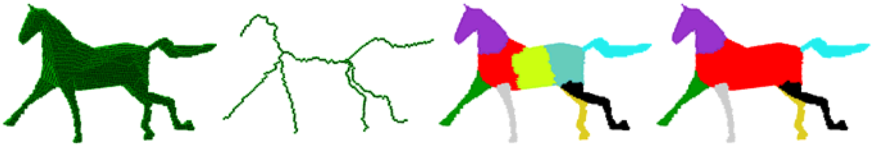

Fig. 5. From left to right, the object "horse", its skeleton, the obtained decomposition before merging non-peripheral simple regions, and the final result

For complex objects, a kernel may be adjacent to more than one non-peripheral simple region. If the visibility criterion is satisfied for all simple regions adjacent to such a kernel, an excessive merging may originate. To solve this problem, we use the following strategy. We associate a multiplicity degree *m* to each kernel, by counting the number of adjacent non-peripheral simple regions satisfying the visibility criterion. If a non-peripheral simple region S_j satisfying the visibility criterion is delimited by kernels having both *m*=1, the three regions are merged. The identity label of the kernel with the largest volume is ascribed to the obtained merged region.

If for S_j one delimiting kernel has *m*=1 while the second delimiting kernel has *m*>1, we first merge S_j with the delimiting kernel with *m*=1. S_j is ascribed the identity label of that kernel. Once merging involving all kernels with *m*=1 has been accomplished, kernels with *m*>1 are considered. If for the inspected kernel K_i the *m* adjacent regions that could undergo merging already carry an identity label typical of kernels, K_i is assigned to the adjacent region with which it shares the largest portion of its surface. As an example see Fig. 6, showing the object "cow", its skeleton, and the decomposition before and after merging. For "cow", one kernel with *m*=2 (shown in black in the decomposition before merging) and two kernels with *m*=1 (shown in red and pink in the decomposition before merging) exist. Both non-peripheral simple regions adjacent to the kernel with *m*=2 satisfy the visibility criterion and are accordingly merged with their second delimiting kernels with *m*=1. Then, the kernel with *m*=2 is merged as described above.

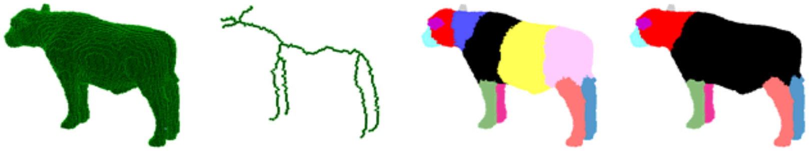

Fig. 6. From left to right, the object "cow", its skeleton, the decomposition before merging (where the kernel shown in black has *m*=2), and the result after merging

As concerns merging of peripheral regions, which can be bumps or simple regions, the visibility criterion is integrated by a further condition taking into account the volume of the object parts. Let S_j be a peripheral region satisfying the visibility criterion and let K_i be the adjacent kernel. Merging is accomplished if the ratio between the volume (measured as number of voxels) of the region union of S_j and K_i, and the volume of K_i is smaller than an a priori fixed threshold τ (set to 1.2 in this work). Using

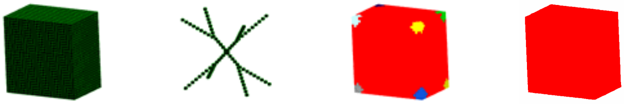

Fig. 7. From left to right, the object “prism”, its skeleton, the decomposition before merging peripheral regions, and the result after merging

the visibility criterion also when dealing with peripheral regions is done to avoid merging elongated and narrow regions that satisfy the condition on volume.

The result of merging peripheral regions is shown in Fig. 7 for the object “prism”.

5 Discussion and Conclusion

We have tested our decomposition procedure on a number of 3D objects taken from publicly available databases [21,22]. A small set of test objects is shown in Fig. 8 together with the corresponding skeletons.

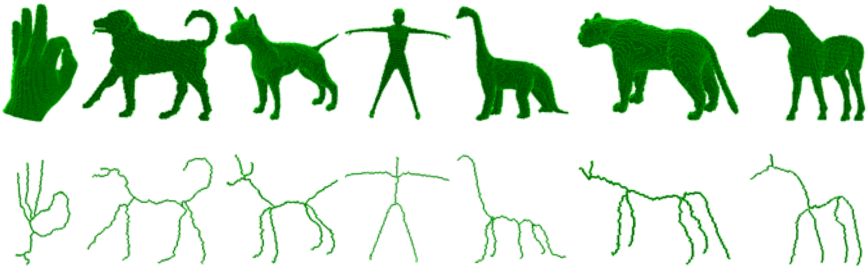

Fig. 8. Test objects and corresponding skeletons

The resulting decompositions before and after merging are shown in Fig.9. The same values for the thresholds θ and τ have been used for all test objects.

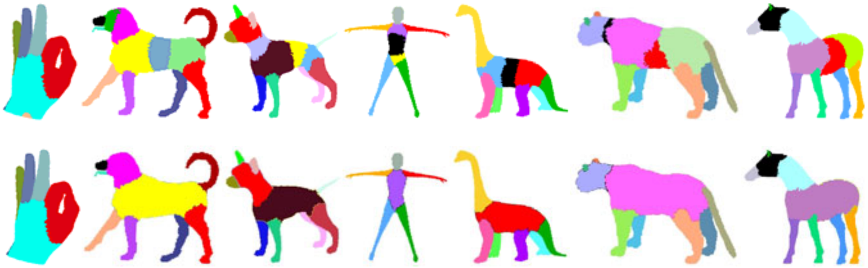

Fig. 9. Decompositions before, top, and after merging, bottom

The values of θ and τ used in this work can be seen as default values. Obviously, threshold values should be tailored to the size of the input objects as well as to the specific problem domain.

We note that if an object is provided in different poses or scales, whichever skeltonization algorithm is used, the skeleton is not guaranteed to have in all cases exactly the same number of branches. Thus, a slightly different partition and, hence, a slightly different decomposition is likely to be obtained. Moreover, we point out that the perceptual significance of the obtained decomposition parts strongly depends on the way skeleton partition has been carried on. In this respect, one could argue that a straightforward partition of the skeleton could be used, where only two types of components are taken into account, namely components consisting of branch points and components consisting of simple curves constituted by the remaining skeleton voxels. However, if this partition is used as input to the recovery process, a perceptual dominance of the regions recovered by the simple curves is obtained at the expenses of the regions reconstructed by the branch points. As a result, the surfaces separating adjacent parts do not cut the object in correspondence with the main concavities along the boundary of the object, but intrude significantly in the object. See Fig. 10 showing the two decompositions obtained for the object “hand”, when starting from the above straightforward skeleton partition and from our partition.

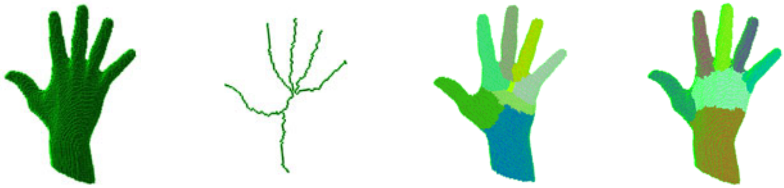

Fig. 10. From left to right, the object “hand”, its skeleton, the decomposition obtained by a straightforward skeleton partition, and the decomposition originated by our skeleton partition

The introduction of the notion of complex sets in the skeleton partition plays a crucial role in obtaining a perceptually significant decomposition. In fact, the kernels are not just coinciding with the union of the balls associated with the branch points, but are obtained by applying the reverse distance transformation to the skeletal voxels included in the zones of influence of the branch points. Thus, kernels are significantly larger than the union of the balls associated only to the branch points and, due to the criterion adopted to manage overlapping, the separations between kernels and the adjacent simple regions are not biased towards the innermost part of the object.

Another important feature of our partition scheme is the fact that some skeleton subsets belonging to a complex set do not participate in the recovery of the corresponding kernel. Each of these subsets is constituted by the voxels linking an end point, taken as single point in the partition, to a branch point. If the linking voxels participate to the recovery of the kernel, the bump generated by the corresponding single point would be a part characterized by a very small volume so that the perceptual relevance of the bump would not be enhanced. In turn, if the linking voxels are removed from the complex set, but are taken together with the end point to constitute a peripheral simple curve, the same problem illustrated in the example in Fig. 10 would occur. Thus, we keep the linking voxels in the complex set, but do not allow them to participate to the recovery process.

The method is easy to implement, has a limited computational cost and produces results in agreement with human intuition.

References

1. Palmer, S.E.: Hierarchical structure in perceptual representation. Cognitive Psychology 9, 441–474 (1977)
2. Marr, D., Nishihara, H.K.: Representation and recognition of three-dimensional shapes. Proc. Royal Society of London: Series B 200, 269–294 (1978)
3. Hoffman, D.D., Richards, W.A.: Parts of recognition. Cognition 18, 65–96 (1984)
4. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review 94, 115–147 (1987)
5. Cornea, N.D., Silver, D., Yuan, X., Balasubramanian, R.: Computing hierarchical curve-skeletons of 3D objects. The Visual Computer 21(11), 945–955 (2005)
6. Lien, J.-M., Geyser, J., Amato, N.M.: Simultaneous shape decomposition and skeletonization. In: Proc. 2006 ACM Symposium on Solid and Physical Modeling, pp. 219–228 (2006)
7. Reniers, D., Telea, A.: Skeleton-based hierarchical shape segmentation. In: Proc. IEEE Int. Conf. on Shape Modeling and Applications, pp. 179–188 (2007)
8. Serino, L., Sanniti di Baja, G., Arcelli, C.: Object decomposition via curvilinear skeleton partition. In: Proc. ICPR 2010, pp. 4081–4084. IEEE, Los Alamitos (2010)
9. Svensson, S., Sanniti di Baja, G.: Using distance transforms to decompose 3D discrete objects. Image and Vision Computing 20, 529–540 (2002)
10. Zhang, X., Liu, J., Jaeger, M., Li, Z.: Volume decomposition for hierarchical skeletonization. Int. J. Virtual Reality 8(1), 89–97 (2009)
11. de Goes, F., Goldenstein, S., Velho, L.: A hierarchical segmentation of articulated bodies. Computer Graphics Forum 27(5), 1349–1356 (2008)
12. Liu, R., Zhang, H.: Segmentation of 3D meshes through spectral clustering. In: Proc. 12th Pacific Conf. on Computer Graphics and Applications, pp. 298–305 (2004)
13. Huang, Q.-X., Wicke, M., Adams, B., Guibas, L.: Shape decomposition using modal analysis. Computer Graphics Forum 28(2), 407–416 (2009)
14. Bischoff, S., Kobbelt, L.: Ellipsoid decomposition of 3D models. In: Proc. Int. Symp. 3D Data Processing Visualization and Transmission, pp. 480–488 (2002)
15. Mortara, M., Patanè, G., Spagnuolo, M., Falcidieno, B., Rossignac, J.: Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies. In: Proc. 9th ACM Symp. on Solid Modeling and Applications, pp. 339–344 (2004)
16. Borgefors, G.: On digital distance transform in three dimensions. CVIU 64(3), 368–376 (1996)
17. Blum, H.: Biological shape and visual science. J. Theor. Biol. 38, 205–287 (1973)
18. Siddiqi, K., Pizer, S.M. (eds.): Medial Representations: Mathematics, Algorithms and Applications. Springer, Heidelberg (2008)
19. Arcelli, C., Sanniti di Baja, G., Serino, L.: Distance driven skeletonization in voxel images. IEEE Trans. PAMI, `http://doi.ieeecomputersociety.org/10.1109/TPAMI.2010.140`
20. Borgefors, G., Sanniti di Baja, G.: Analyzing non-convex 2D and 3D patterns. CVIU 63(1), 145–157 (1996)
21. AIM@SHAPE Shape Repository, `http://shapes.aimatshape.net/viewmodels.php`
22. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton Shape Benchmark. Shape Modeling International, Genova, Italy (June 2004)