# Retrieving Articulated 3-D Models Using Medial Surfaces and their Graph Spectra

Juan Zhang[1], Kaleem Siddiqi[1], Diego Macrini[2], Ali Shokoufandeh[3], and Sven Dickinson[2]

[1]McGill University, School of Computer Science & Centre for Intelligent Machines
[2]University of Toronto, Department of Computer Science
[3]Drexel University, Department of Computer Science

**Abstract.** We consider the use of medial surfaces to represent symmetries of 3-D objects. This allows for a qualitative abstraction based on a directed acyclic graph of components and also a degree of invariance to a variety of transformations including the articulation and deformation of parts. We demonstrate the use of this representation for both indexing and matching 3-D object models. Our formulation uses the geometric information associated with each node along with an eigenvalue labeling of the adjacency matrix of the subgraph rooted at that node. We present comparative results against the techniques of shape distributions [17] and harmonic spheres [12] on a database of 320 models representing 13 object classes. The results demonstrate that medial surface based graph matching significantly outperforms these techniques for objects with articulating parts.

*Keywords* 3-D model matching, indexing, medial surfaces, graph spectra.

## 1 Introduction

With an explosive growth in the number of 3-D object models stored in web repositories and other databases, the graphics community has begun to address the important and challenging problem of 3-D object retrieval and matching, a problem which traditionally falls in the domain of computer vision research. Recent advances include query-based search engines which employ promising measures including spherical harmonic descriptors [12] and shape distributions [17]. Such systems can yield impressive results on databases including hundreds of 3-D models, in a matter of a few seconds.

Thus far the emphasis in the computer graphics community has broadly been on the use of qualitative measures of shape that are typically global. Such measures are robust in the sense that they can deal with noisy and imperfect models, and at the same time are simple enough so that efficient algorithmic implementations can be sought. However, an inevitable cost is that such measures are inherently coarse, and are sensitive to deformations of objects or their parts. As a motivating example, consider the 3-D models in Fig. 1. These four exemplars of an object class were created by articulations of parts and changes of pose. For such examples, the very notion of a center of mass or a rigid reference point[1], which is crucial for the computation of descriptions such as shape histograms (sectors or shells) [3] or spherical extent functions [30], can be nonintuitive

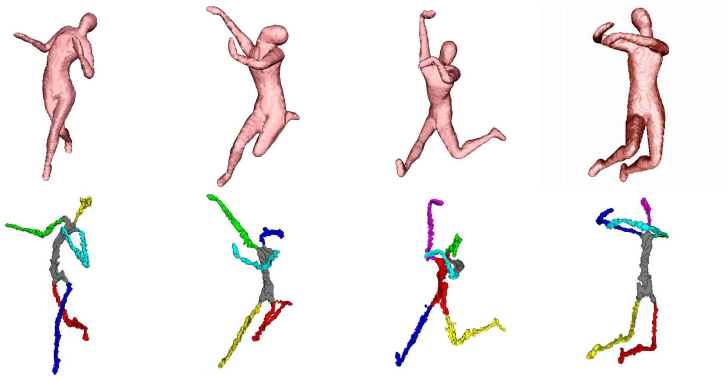

**Fig. 1.** Exemplars of the object class "human" created by changes in pose and articulations of parts (top row). The medial surface (or 3-D skeleton) of each is computed using the algorithm of [27] (bottom row). The medial surface is automatically partitioned into distinct parts, each shown in a different color.

and arbitrary. In fact, the centroid of such models may actually lie in the background. To complicate matters, it is unclear how to obtain a global alignment of such models. As well, measures based on reflective symmetries [11], and signatures based on 3-D moments [7] or chord histograms [17] are not invariant under such transformations.

The computer vision community has grappled with the problem of *generic* level object recognition by suggesting representations based on volumetric parts, including generalized cylinders, superquadrics and geons [5, 16, 19, 4]. Such approaches build a degree of robustness to deformations and movement of parts, but their representational power is limited by the vocabulary of geometric primitives that are selected. Motivated in part by such considerations there have been attempts to encode 3D shape information using probabilistic descriptors. These allow intrinsic geometric information to be captured by low dimensional signatures. An elegant example of this is the geodesic shape distribution of [9] where information theoretic measures are used to compare probability distributions representing 3D object surfaces. In the domain of graph theory there have also been attempts to address the problem of 3D shape matching using representations based on Reeb graphs [24, 10]. These allow for topological properties to be captured, at least in a coarse sense.

An alternative approach is to use 3-D medial loci (3-D skeletons), obtained by considering the locus of centers of maximal inscribed spheres along with their radii [6]. As pointed out by Blum, this offers the advantage that a graph of parts can be inferred from the underlying local mirror symmetries of the object. To motivate this idea, consider once again the human forms of Fig. 1. A medial surface-based representation (bottom row) provides a natural decomposition, which is largely invariant to the articulation and bending of parts.

In this article, we build on a recent technique to compute medial surfaces [27] by proposing an interpretation of its output as a directed acyclic graph (DAG) of parts. We then use refinements of algorithms based on graph spectra [26] to tackle the prob-

lems of indexing and matching 3-D object models. These and related algorithms have already shown promise in the computer vision community for generic level view-based object indexing and matching using 2-D skeletal graphs [28, 25, 18, 22]. They have also been demonstrated in the context of matching 3-D object models with tubular parts, using a centerline approximation of the 3-D skeleton [29]. We demonstrate their significant potential for medial surface-based 3-D object retrieval with experimental results on a database of 320 models representing 13 object classes, including exemplars of both rigid objects and ones with significant articulation of parts. Comparative results using the information retrieval notion of *precision versus recall* demonstrate that this method significantly outperforms the popular techniques of shape distributions [17] and harmonic spheres [12] for objects with articulating parts. To our knowledge these are the first comprehensive empirical results on the use of medial surfaces and their graph spectra in the context of 3-D object model retrieval and indexing.

## 2 Medial Surfaces and DAGs

Recent approaches for computing 3-D skeletons include the *power crust* algorithm [2], the *shock scaffold* [13] and *average outward flux-based* skeletons [27]. The first two methods have the advantage that they can be employed on input data in the form of points sampled from an object's surface, and theoretical guarantees on the quality of the results can be provided. Unfortunately, automatic segmentation of the resulting skeletons remains a challenge. The last method assumes that objects have first been voxelized, and this adds a computational burden. However, once this is done the limiting behavior of the average outward flux of the Euclidean distance function gradient vector field can be used to characterize 3-D skeletal points. We choose to employ this latter method since it has the advantage that the digital classification of [15] allows for the taxonomy of generic 3-D skeletal points [8] to be interpreted on a rectangular lattice, leading to a graph of parts.

Under the assumption that the initial model is given in triangulated form, we begin by scaling all the vertices so that they fall within a rectangular lattice of fixed dimension and resolution. We then sub-divide each triangle to generate a dense intersection with this lattice, resulting in a binary (voxelized) 3-D model. The average outward flux of the Euclidean distance function's gradient vector field is computed through unit spheres centered at each rectangular lattice point, using Algorithm 1. This quantity has the property that it approaches a negative number at skeletal points and goes to zero elsewhere [27], and thus can be used to drive a digital thinning process, for which an efficient implementation is described in Algorithm 2. This thinning process has to be implemented with some care, so that the topology of the object is not changed. This is done by identifying each *simple* or removable point $\mathbf{x}$, for which a characterization based on the 26-neighborhood of each lattice point $\mathbf{x}$ is provided in [15]. With O being the set of points in the interior of the voxelized object and $\mathrm{N}_{26}^{*}$ being the 26-neighborhood of $\mathbf{x}$, not including $\mathbf{x}$ itself, this characterization is based on two numbers:

1. $\overline{C}^{*}$: the number of 26-connected components 26-adjacent to $\mathbf{x}$ in $\mathrm{O} \cap \mathrm{N}_{26}^{*}$, and
2. $\overline{C}$: the number of 6-connected components 6-adjacent to $\mathbf{x}$ in $\overline{\mathrm{O}} \cap \mathrm{N}_{18}$.

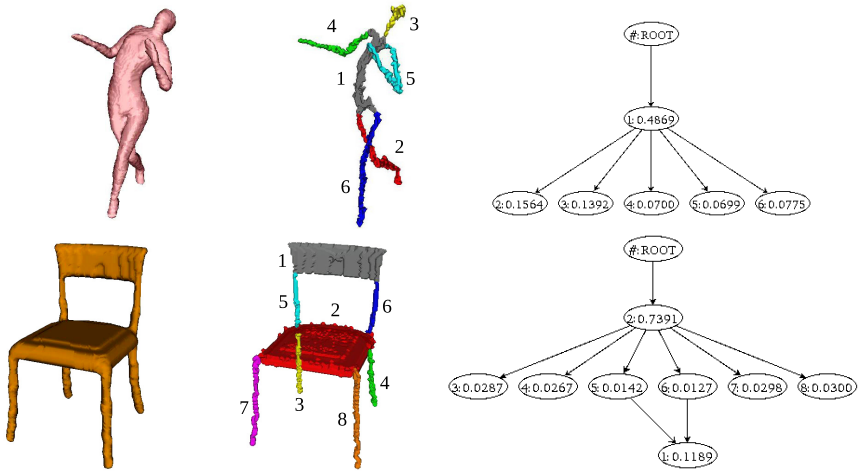


**Fig. 2.** A voxelized human form and chair (left) and their segmented medial surfaces (middle). A hierarchical interpretation of the medial surface, using a notion of part saliency, leads to a directed acyclic graph DAG (right). The nodes in the DAGs have labels corresponding to those on the medial surface, and the saliency of each node is also shown.

---

**Algorithm 1:** Average Outward Flux.

---

**Data** : Voxelized 3-D Object Model.
**Result** : Average Outward Flux Map.
Compute the Euclidean distance transform $D$ of the model ;
Compute the gradient vector field $\nabla D$;
Compute the average outward flux of $\nabla D$:
For (each point $\mathbf{x}$) $\mathrm{AOF}(\mathbf{x}) = \frac{1}{26}\sum_{i=1}^{26} < \hat{\mathbf{N}}_\mathbf{i}, \nabla D(\mathbf{x}_i) >$;
(where $\mathbf{x}_i$ is a 26-neighbor of $\mathbf{x}$ in 3-D and $\hat{\mathbf{N}}_\mathbf{i}$ is the outward normal at $\mathbf{x}_i$ of the unit sphere centered at $\mathbf{x}$)

---

It can be shown that a digital point $\mathbf{x}$ is *simple* if $C^*(\mathbf{x}) = 1$ and $\bar{C}(\mathbf{x}) = 1$.

The taxonomy of generic 3-D skeletal points in the continuum, i.e., those which are stable under small perturbations of the object, is provided in [8]. Using the notation $A_n^k$, where $n$ denotes the number of points of contact of the maximal inscribed sphere with the surface and $k$ the order of these contacts, the taxonomy includes: 1) $A_1^2$ points which form a smooth medial manifold, 2) $A_3$ points which correspond to the rim of a medial manifold, 3) $A_1^3$ points which represent the intersection curve of three medial manifolds, 4) an $A_1^4$ point at the intersection of four $A_1^3$ curves, and 5) an $A_1A_3$ point at the intersection between an $A_3$ curve and an $A_1^3$ curve.

It is clear from this classification that 3-D skeletons are essentially comprised of medial manifolds, their rims and intersection curves, and this is why we refer to this as a

**Algorithm 2:** Topology Preserving Thinning.

```
Data   : 3-D Object Model, Average Outward Flux Map.
Result : 3-D Skeleton (Medial Surface).
for (each point x on the boundary of the object) do
    if (x is simple) then
        insert(x, maxHeap) with AOF(x) as the sorting key for insertion;
while (maxHeap.size > 0) do
    x = HeapExtractMax(maxHeap);
    if (x is simple) then
        if (x is an end point) and (AOF(x) < Thresh) then
            mark x as a medial surface (end) point;
        else
            Remove x;
            for (all neighbors y of x) do
                if (y is simple) then
                    insert(y, maxHeap) with AOF(y) as the sorting key for insertion;
```

*medial surface* representation. As shown in [15], the numbers $C^*$ and $\bar{C}$ can also be used to classify surface points, rim points, junction points and curve points on a rectangular lattice. These results are summarized in Table 1. This suggests the following 3-step approach for segmenting the (voxelized) medial surface into a set of connected parts:

1. Identify all manifolds comprised of 26-connected surface points and border points.
2. Use junction points to separate these manifolds, but allow junction points to belong to all manifolds that they connect.
3. Form connected components with the remaining curve points, and consider these as parts as well.

This process of automatic skeletonization and segmentation is illustrated for two object classes, a chair and a human form, in Fig. 2.

We now propose an interpretation of the segmented medial surface as a directed acyclic graph (DAG), where we shall treat each component as a node. This will in turn allow the subsequent matcher and indexer to cope with both changes in part structure, as reflected by connectivity in the graph, as well as changes in part shape, as reflected by the geometric information associated with each node. We begin by introducing a notion of *saliency* which captures the relative importance of each component. Consider that the envelope of maximal inscribed spheres of appropriate radii placed at all skeletal points reconstructs the original object's volume [6]. The contribution of each component to the overall volume can thus be used as a measure of its significance. Since the spheres associated with adjacent components can overlap, an objective measure of component $j$'s saliency is given by:

$$\mathrm{S}aliency_j = \frac{\mathrm{V}oxels_j}{\sum_{i=1}^{N} \mathrm{V}oxels_i},$$

| $C$ | $C^*$ | TYPE |
|---|---|---|
| 0 | any | interior point |
| any | 0 | isolated point |
| 1 | 1 | border (simple) point |
| 1 | 2 | curve point |
| 1 | $> 2$ | curves junction |
| 2 | 1 | surface point |
| 2 | $> 2$ | surface-curve(s) junction |
| $> 2$ | 1 | surfaces junction |
| $> 2$ | $\geq 2$ | surfaces-curves junction |

**Table 1.** The topological classification of [15].

where $N$ is the number of components and V*oxels*$_i$ is the number of voxels *uniquely* reconstructed by component $i$. We propose the following construction of a DAG, using each component's saliency. Consider the most salient component as the root node (level 0), and place components to which it is connected as nodes at level 1. Components to which these nodes are connected are placed at level 2, and this process is repeated in a recursive fashion until all nodes are accounted for. The graph is completed by drawing edges between all pairs of connected nodes, in the direction of increasing levels, hence avoiding the occurrence of any cycles. However, to allow for 3-D models comprised of disconnected parts we introduce a single dummy node as the parent of all DAGs for a 3-D model.

This process is illustrated in Fig. 2 (right column) for the human and chair models, with the saliency values shown within the nodes. Note how this representation captures the intuitive sense that the human is a torso with attached limbs and a head, a chair is a seat with attached legs and a back, etc. Our DAG representation of the medial surface is quite different than the graph structure that follows from a direct use of the taxonomy of 3-D skeletal points in the continuum [8]. Our motivation is to be able to exploit the hierarchical structure indexing and matching algorithms reported in [28, 26].

## 3 Indexing

A linear search of the 3-D model database, i.e., comparing the query 3-D object model to each 3-D model and selecting the closest one, is inefficient for large databases. An indexing mechanism is therefore essential to select a small set of candidate models to which the matching procedure is applied. When working with hierarchical structures, in the form of DAGs, indexing is a challenging task, and can be formulated as the fast selection of a small set of candidate model graphs that share a subgraph with the query. But how do we test a given candidate without resorting to subgraph isomorphism and its intractability? The problem is further compounded by the fact that due to perturbation and noise, no significant isomorphisms may exist between the query and the (correct) model. Yet, at some level of abstraction, the two structures (or two of their substructures) may be quite similar. Thus, our indexing problem can be reformulated as finding model (sub)graphs whose structure is *similar* to the query (sub)graph.

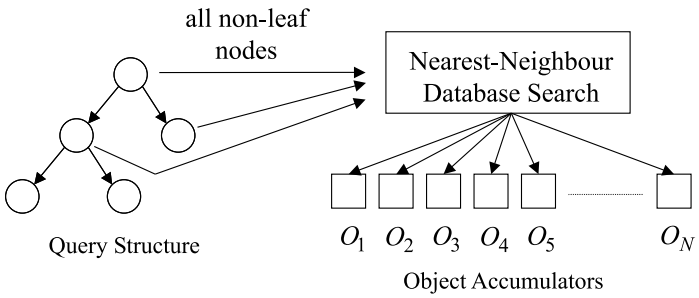


**Fig. 3.** Indexing Mechanism. Each non-trivial node (whose TSV encodes a topological abstraction of the subgraph rooted at the node) votes for models sharing a structurally similar subgraph. Models receiving strong support are candidates for a more comprehensive matching process.

Choosing the appropriate level of abstraction with which to characterize a DAG is a challenging problem. We seek a description that, on the one hand, provides the low dimensionality essential for efficient indexing, while on the other hand, is rich enough to prune the database down to a tractable number of candidates. In recent work [26], we draw on the eigen-space of a graph to characterize the topology of a DAG with a low-dimensional vector that will facilitate an efficient nearest-neighbor search in a database. The eigenvalues of a graph's adjacency matrix encode important structural properties of the graph, characterizing the degree distribution of its nodes. Moreover, we have shown that the magnitudes of the eigenvalues are stable with respect to minor perturbations of graph structure due to, for example, noise, segmentation error, or minor within-class structural variation. For every rooted directed acyclic subgraph (i.e., part) of the original DAG, we compute a function of the eigenvalues of the subgraph's antisymmetric $\{0, 1, -1\}$ node-adjacency matrix which yields a low-dimensional *topological signature vector* (TSV) encoding of the "shape" of the subgraph. Details of the TSV, along with an analysis of its stability, can be found in [26].

Indexing now amounts to a nearest-neighbor search in a model database, as shown in Fig. 3. The TSV of each non-leaf node (the root of a graph "part") in each model DAG defines a vector location in a low-dimensional Euclidean space (the model database) at which a pointer to the model containing the subgraph rooted at the node is stored. At indexing time, a TSV is computed for each non-leaf node, and a nearest-neighbor search is performed using each "query" TSV. Each TSV "votes" for nearby "model" TSVs, thereby accumulating evidence for models that share the substructure defined by the query TSV. Indexing could, in fact, be accomplished by indexing solely with the root of the entire query graph. However, in an effort to accommodate large-scale perturbation (which corrupts all ancestor TSVs of a perturbed subgraph), indexing is performed locally (using all non-trivial subgraphs, or "parts") and evidence combined. The result is a small set of ranked model candidates which are verified more extensively using the matching procedure described next.

## 4 Matching

Each of the top-ranking candidates emerging from the indexing process must be verified to determine which is most similar to the query. If there were no noise our problem could be formulated as a graph isomorphism problem for vertex-labeled graphs. With limited noise, we would search for the largest isomorphic subgraph between query and model. Unfortunately, with the presence of significant noise, in the form of the addition and/or deletion of graph structure, large isomorphic subgraphs may simply not exist. This problem can be overcome by using the same eigen characterization of graph structure we use as the basis of our indexing mechanism [28].

As we know, each node in a graph (query or model) is assigned a TSV, which reflects the underlying structure in the subgraph rooted at that node. If we simply discarded all the edges in our two graphs, we would be faced with the problem of finding the best correspondence between the nodes in the query and the nodes in the model; two nodes could be said to be in close correspondence if the distance between their TSVs (and the distance between their domain-dependent node labels) was small. In fact, such a formulation amounts to finding the maximum cardinality, minimum weight matching in a bipartite graph spanning the two sets of nodes. In a modification of Reyner's algorithm [20], we combine the above bipartite matching formulation with a greedy, best-first search in a recursive procedure to compute the corresponding nodes in two rooted DAGs which, in turn, yields an overall similarity measure that can be used to rank the candidate. Details of the algorithm can be found in [28, 14].

### 4.1 Node Similarity

The above matching algorithm requires a node similarity function that compares the shapes of the 3-D parts associated with two nodes. A variety of the measures used in the literature as signatures for indexing entire 3-D models could be used to compute similarities between two parts (nodes) [17, 3, 30, 7, 11]. Some care would of course have to be taken in the implementation of methods which require a form of global alignment. We have opted for a much simpler measure, which is based on the use of a mean curvature histogram.

First, consider the volumetric part that a node $i$ represents, along with its Euclidean distance function $D$. At any point within this volume, the mean curvature of the iso-distance level set is given by $\mathrm{div}(\frac{\nabla D}{||\nabla D||})$. On a voxel grid with unit spacing the observable mean curvatures are in the range $[-1,1]$. We compute a histogram of the mean curvature over all voxels in the volumetric part, over this range, using a fixed number of bins $N$. A mean curvature histogram vector $\hat{M}_i$ is then constructed with entries representing the fraction of total voxels in each bin. The similarity between two nodes $i$ and $j$ is then given by the following measure, which uses the sum of squared distances between the corresponding entries $k$ of each node's mean curvature histogram vector:

$$Similarity(i,j) = [1 - \underbrace{\sqrt{\sum_{k=1}^{N} [\hat{M}_i(k) - \hat{M}_j(k)]^2}}_{Distance(i,j)}].$$

By construction, this similarity function is in the interval $[0, 1]$. This measure could be further modified to take into account overall part sizes. In our experiments we choose not to do this since our object models have undergone a global size normalization.

## 5 Experimental Results

In order to test the power of our indexing and matching algorithms using medial surface-based DAGs, we have considered using the Princeton Shape Benchmark [23]. This standardized database, which contains 1,814 3-D object models organized by class, is an effective one for comparing the performance of a variety of methods including those in [11, 17, 3, 30, 7]. A majority of the models in the database correspond to rigid, man-made objects for which a notion of a centroid applies. The natural objects include a variety of animals (including humans), trees, plants and body parts. However, only a limited number of these have articulated or deformed parts. When such models are present, the precise nature of part articulation typically defines a unique base level category. For example, *animal-biped-human* contains human models which are upright, *animal-biped-human-arms-out* contains similar models with outstretched hands and *animal-biped-human-walking* contains those in a walking pose. Results reported in [23] indicate that a number of global shape descriptors perform suitably at such base levels of classification, but degrade rapidly at coarser levels, e.g., the classification *human*. In the context of *generic* 3-D model retrieval, such coarser levels in fact correspond to the notion of a *basic level* or *entry level* categorization [21, 4], whose exemplars might reflect a variety of complex poses and articulations, such as those seen in Fig. 1. Our matching and indexing algorithms have the potential to work at this more challenging level, because they use intuitive part-based representations.

To demonstrate this, we have constructed our own database adopting some of the models in the Princeton repository, but adding several of our own. Our database includes a total of 320 exemplars taken from several *basic level* object classes (hands, humans, teddy bears, glasses, pliers, tables, chairs, cups, airplanes, birds, dolphins, dinosaurs, four-legged animals, fish). A large number of these models are shown in Fig. 5. We divide these classes into two categories, those with significant part articulation, and those with moderate or no part articulation. In our experiments we merge the categories "four-legged" and "dinosaurs", treating them as a single category "four-limbs"

*Indexing Results:* In order to test our indexing algorithm, which utilizes only the topological structure of medial surface-based DAGs, we carried out two types of experiments. In the first we evaluated percentage recall. For a number of rank thresholds the percentage of models in the database in the same category as a query (not including the query itself) with higher indexing rank, are shown in Fig. 5. The results indicate that on average 70% of the desired models are in the top 80 (25% of 320) ranks. In the second experiment we examine the average ranks according to object classes. For all queries in a class the rank of all other objects in that class is computed. The ranks averaged across that class are shown in Fig. 6. The results indicate that for 9 of the 13 object classes the average rank is in the top 80 (25% of 320). The higher average ranks for the remaining classes are due to the fact that certain categories have similar part decompositions.

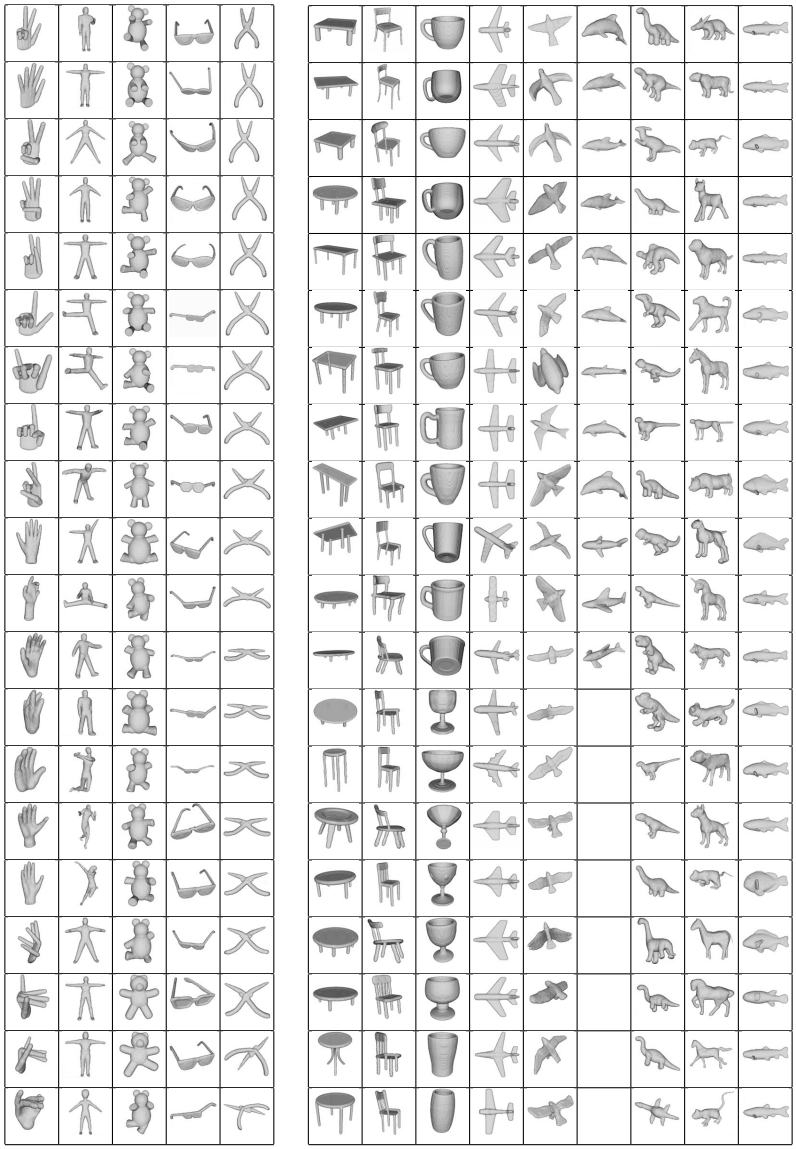


**Fig. 4.** Database Exemplars. 20 members are shown from each of the object classes (with the exception of the class dolphins which has fewer exemplars). Exemplars from classes on the left have significant part articulation of a complexity not seen in the Princeton Shape Benchmark. Note that we treat the dinosaurs and the four-legged animals as members of a single object class "four-limbs".

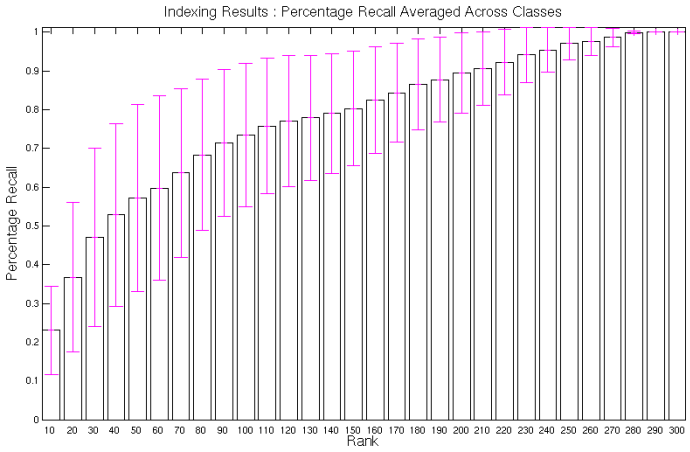


**Fig. 5.** Indexing Results: Percentage Recall. For several rank thresholds, $N = 10, 20, ...$, we plot the percentage of models in the database in the same category as the query (not including the query itself) with indexing rank $\leq N$. The results averaged across all classes are shown along with error bars depicting +/- 1 standard deviation.

In such cases topological structure on its own is not discriminating enough, and part shapes also have to be taken into account.

It should be emphasized that the indexer is a fast screener which can quickly prune the database down to a much smaller set of candidates to which the matcher can be applied. Furthermore, the eigen characterization used to compute the index is also used at matching time, so the same eigen structure calculation is exploited for both steps. The systems against which we evaluate the matcher in the following section run a linear search on the entire database for each query. This approach does not scale well, since the indexing problem is essentially ignored.

*Matching Results:* On a large database we envision running the indexing strategy first to obtain a smaller subset of candidate 3-D models and to match the query only against these. However, given the moderate size of our database we were able to generate the $320 \times 320 = 102,400$ pairs of matches in a matter of 15-20 minutes on a 3.0 GHz desktop PC. We compare the results using medial surfaces (MS) with those obtained using harmonic spheres (HS) [12] and shape distributions (SD) [17]. The pair-wise distances between models using harmonic spheres were obtained using Michael Kazhdan's executable code (http://www.cs.jhu.edu/~ misha) and those using shape distributions were based on our own implementation of the algorithm described in [17]. For both HS and SD we used as input a mesh representation of the bounding voxels of the voxelized model used for MS. The comparisons are performed using the standard information retrieval notion of *recall versus precision*, where curves shifted upwards and to the right indicate superior performance.

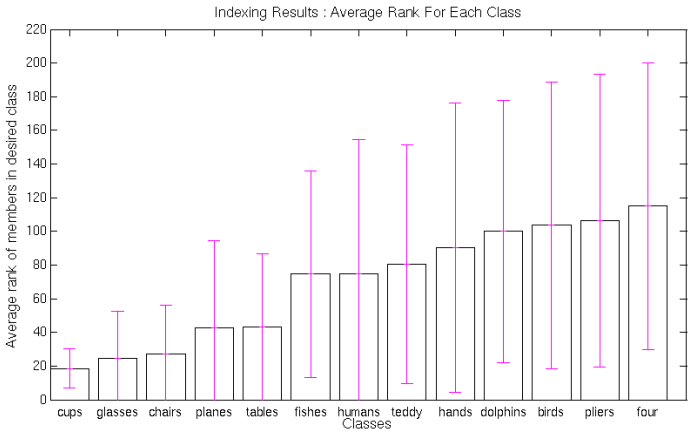


**Fig. 6.** Indexing Results: Average Ranks. For all queries in a class the rank of all other objects in that class are computed. The ranks averaged across that class are shown, along with error bars depicting +/- 1 standard deviation.

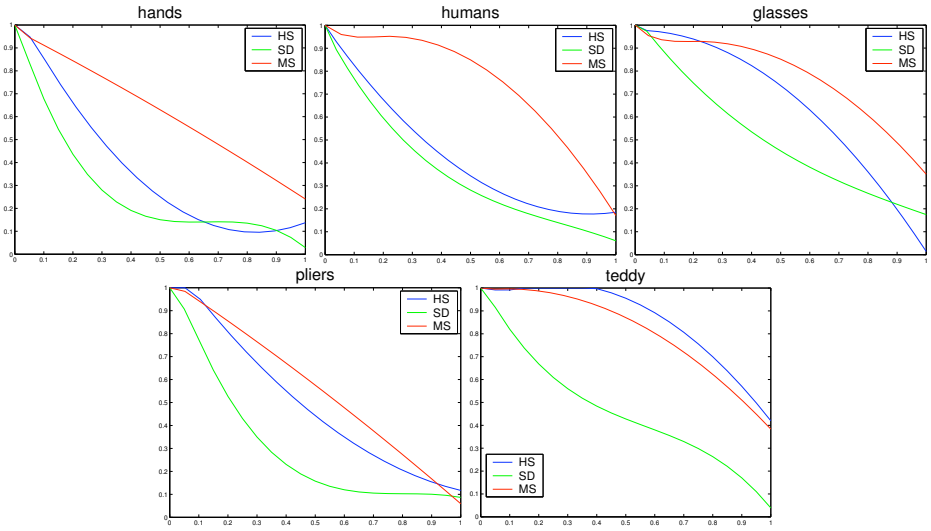


**Fig. 7.** Precision (y axis) versus Recall (x axis): Objects with articulating parts. The results using medial surfaces (MS) are shown in red, those using harmonic spheres (HS) are shown in blue and those using shape distribution (SD) are shown in green. The results obtained using MS are superior for all categories with the exception of the category “teddy” for which both HS and MS give excellent results.

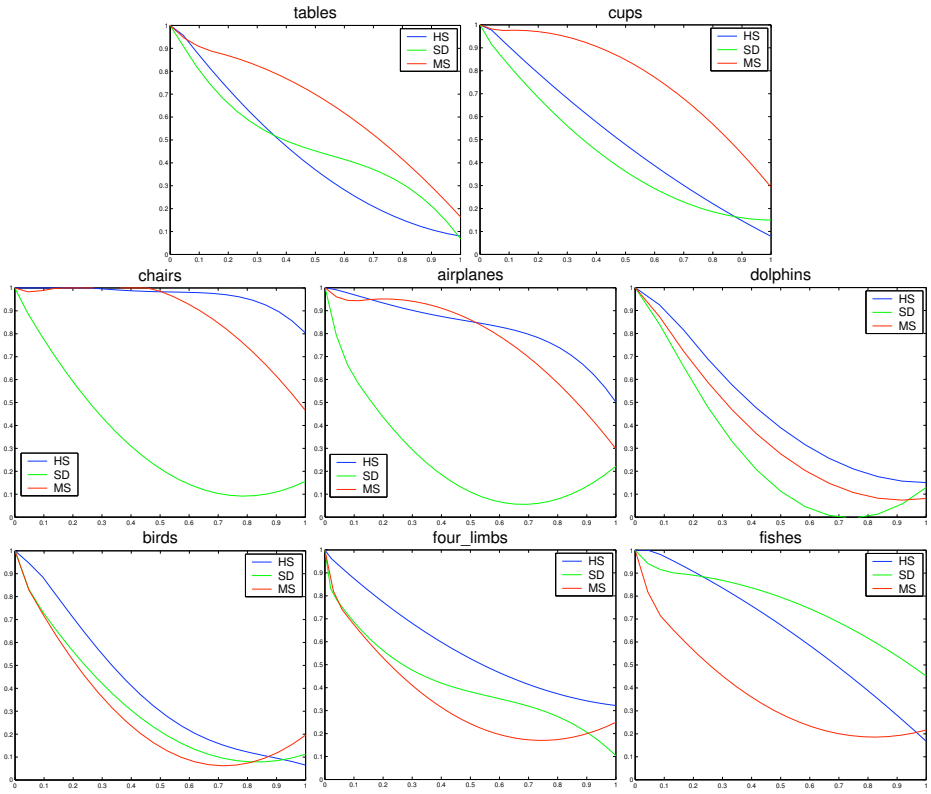


**Fig. 8.** Precision (y axis) versus Recall (x axis): Objects with moderate or no articulation. The results using medial surfaces (MS) are shown in red, those using harmonic spheres (HS) are shown in blue and those using shape distribution (SD) are shown in green. For categories in the top row MS gives superior results. For categories in the middle row HS gives slightly better results than MS, but both are superior to SD. For categories in the third row the results are comparable for birds, but for four-limbs and fishes, both HS and SD out perform MS.

The results for objects with articulating parts are presented in Fig. 7. For the category "teddy" both MS and HS give excellent results and for all other categories MS outperforms the other two techniques. Fig. 8 shows the results for objects with moderate or no part articulation. For categories in the top row MS gives superior results. For categories in the middle row HS gives slightly better results than MS, but both are significantly superior to SD. For categories in the third row the results are comparable for birds, but for four-limbs and fishes, both HS and SD out perform MS. In Table 5 we show the average similarity scores using MS, organized by object class. Red and blue boxes are drawn, respectively, around the two highest similarity scores. In all cases the highest score coincides with the correct object class. Overall these results demonstrate the significant potential of medial surface based representations and their graph spectra for *generic* level 3-D model retrieval, despite substantial articulation of parts.

| Instance | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | .61 | .37 | .00 | .45 | .23 | .20 | .02 | .02 | .10 | .00 | .09 | .16 | .26 |
| | .37 | .38 | .00 | .21 | .25 | .18 | .12 | .10 | .18 | .02 | .18 | .23 | .25 |
| | .00 | .00 | .51 | .29 | .17 | .15 | .07 | .03 | .00 | .15 | .02 | .00 | .07 |
| | .45 | .21 | .29 | .64 | .34 | .23 | .04 | .04 | .00 | .01 | .05 | .04 | .28 |
| | .23 | .25 | .17 | .34 | .43 | .24 | .16 | .15 | .12 | .04 | .22 | .06 | .19 |
| | .20 | .18 | .15 | .23 | .24 | .28 | .20 | .22 | .14 | .07 | .26 | .05 | .14 |
| | .02 | .12 | .07 | .04 | .16 | .20 | .51 | .46 | .37 | .08 | .45 | .00 | .09 |
| | .02 | .10 | .03 | .04 | .15 | .22 | .46 | .53 | .29 | .03 | .47 | .02 | .06 |
| | .10 | .18 | .00 | .00 | .12 | .14 | .37 | .29 | .58 | .04 | .31 | .02 | .23 |
| | .00 | .02 | .15 | .01 | .04 | .07 | .08 | .03 | .04 | .48 | .08 | .15 | .07 |
| | .09 | .18 | .02 | .05 | .22 | .26 | .45 | .47 | .31 | .08 | .56 | .02 | .12 |
| | .16 | .23 | .00 | .04 | .06 | .05 | .00 | .02 | .02 | .15 | .02 | .71 | .21 |
| | .26 | .25 | .07 | .28 | .19 | .14 | .09 | .06 | .23 | .07 | .12 | .21 | .40 |

**Table 2.** Average Matching Results Using MS. Each object in the database is matched against all the other objects in the database. Each cell shows the average similarity between objects selected from two fixed object classes. In each row red and blue boxes are drawn, respectively, around the two highest average similarity scores. In all cases the highest score coincides with the correct object class. In most cases there is also a very significant difference between the top two average similarity scores.

## 6 Conclusion

We advance the state-of-the-art in 3-D object model retrieval by: 1) introducing a modification of a Euclidean distance function-based method for computing and segmenting medial surfaces, 2) proposing a DAG representation of the medial surface which captures a notion of part saliency, 3) building on algorithms in the computer vision literature to address the problem of 3-D model indexing and matching in a uniform framework and 4) presenting indexing and matching results on a database of object models organized according to an *entry* level of categorization, with categories having significant part articulation. Whereas all the pieces of this system have been developed in past work, putting them together and demonstrating them in the context of 3-D model retrieval with comparative results against competing methods has been the focus of this article.

The major current limitations of our work include: 1) the assumption that the original object models can be voxelized, 2) the coarse nature of the part similarity measure based on mean curvature histograms, and 3) the assumption that objects with complex part topologies can yield stable graph structures using medial surface decompositions on a digital lattice. First, it is feasible to "patch" models with a few missing triangles, so that voxelization becomes possible. It might also be fruitful to explore Voronoi methods for computing medial surface-based DAGs that could in principle be applied directly to point clouds, provided that the sampling density is high enough [2] or to use the shock scaffold technique [13]. However, for models with incomplete surfaces and large holes, and hence no well defined notion of an interior and an exterior, medial surface-based DAGs would not be appropriate. With regard to the second limitation, indeed we expect that the performance of graph theoretic algorithms for comparing medial surface based representations will improve with more discriminating part similarity measures, and any one of a number suggested in the literature can be investigated. Finally, the third concern (as exemplified by the poorer results on the four-limbed animals) points to some limitations of the current representation. It is well known that certain regions of the medial locus are less stable than others, such as Blum's ligatures [6]. Thus, there is more work to be done both in the direction of developing robust techniques for segmenting 3D skeletons as well as in selecting its stable manifolds for building representations.

## References


1. H. Alt, O. Aichholzer, and G. Rote. Matching Shapes With a Reference Point. In *Proceedings of the Tenth Annual Symposiusm On Computational Geometry*, pages 85–92, 1994.
2. N. Amenta, S. Choi, and R. Kolluri. The Power Crust, Unions of Balls, and the Medial Axis Transform. *Computational Geometry: Theory and Applications*, 19(2):127–153, 2001.
3. M. Ankerst, G. Kastenmüller, H. Kriegel, and T. Seidl. 3-D Shape Histograms for Similarity Search and Classification in Spatial Databases. In *Advances in Spatial Databases, 6th International Symposium*, volume 18, pages 700–711, 1999.
4. I. Biederman. Recognition–By–Components: A Theory of Human Image Understanding. *Psychological Review*, 94(2):115–147, 1987.
5. T. O. Binford. Visual Perception by Computer. In *IEEE Conference on Systems and Control*, December 1971.
6. H. Blum. Biological Shape and Visual Science. *Journal of Theoretical Biology*, 38:205–287, 1973.
7. M. Elad, A. Tal, and S. Ar. Content Based Retrieval of VRML Objects- An Iterative and Interactive Approach. In *6th Europgraphics Workshop on Multimedia*, pages 107–118, Manchester, UK, 2001.
8. P. J. Giblin and B. B. Kimia. A formal Classification of 3D Medial Axis Points and Their Local Geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):238–251, February 2004.
9. A. B. Hamza and H. Krim. Geodesic Object Representation and Recognition. In *Proceedings of DGCI*, volume LNCS 2886, pages 378–387, 2003.
10. M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes. In *Proceedings of ACM SIGGRAPH*, pages 203–212, 2001.
11. M. Kazhdan, B. Chazelle, D. Dobkin, T. Funkhouser, and S. Rusinkiewicz. A Reflective Symmetry Descriptor for 3-D Models. *Algorithmica*, 38(1):201–225, October 2003.

12. M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation Invariant Spherical Harmonic Representation of 3D Shape Descriptors. In *Symposium on Geometry Processing*, jun 2003.
13. F. F. Leymarie and B. B. Kimia. Computation of the Shock Scaffold for Unorganized Point Clouds in 3D. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 821–827, Madison, Wisconsin, 2003.
14. D. Macrini. *Indexing and Matching for View-Based 3-D Object Recognition Using Shock Graphs*. PhD thesis, University of Toronto, 2003.
15. G. Malandain, G. Bertrand, and N. Ayache. Topological Segmentation of Discrete Surfaces. *International Journal of Computer Vision*, 10(2):183–197, 1993.
16. D. Marr and K. H. Nishihara. Representation and Recognition of the Spatial Organization of Three Dimensional Structure. *Proceedings of the Royal Society of London*, B 200:269–294, 1978.
17. R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape Distributions. *ACM Transactions on Graphics*, 21(4):807–832, oct 2002.
18. M. Pellilo, K. Siddiqi, and S. W. Zucker. Matching Hierarchical Structures Using Association Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1105–1120, 1999.
19. A. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28:293–331, 1986.
20. S. W. Reyner. An Analysis of a Good Algorithm for the Subtree Problem. *SIAM J. Comput.*, 6:730–732, 1977.
21. E. Rosch. Principles of Categorization. In *Cognition and Categorization*. L. Erlbaum Associates, 1978.
22. T. Sebastian, P. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:551–571, 5 2004.
23. P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The Princeton Shape Benchmark. In *Shape Modeling International*, Genova, Italy, June 2004.
24. Y. Shinagawa, T. L. Kunii, and Y. L. Kergosien. Surface Coding Based on Morse Theory. *IEEE Transactions On Computer Graphics and Applications*, 11(5):66–78, 1991.
25. A. Shokoufandeh, S. J. Dickinson, K. Siddiqi, and S. W. Zucker. Indexing Using a Spectral Encoding of Topological Structure. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 491–497, Fort Collins, CO, June 1999.
26. A. Shokoufandeh, D. Macrini, S. Dickinson, K. Siddiqi, and S. Zucker. Indexing Hierarchical Structures Using Graph Spectra. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 27(7), 2005.
27. K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker. Hamilton-Jacobi Skeletons. *International Journal of Computer Vision*, 48(3):215–231, 2002.
28. K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock Graphs and Shape Matching. *International Journal of Computer Vision*, 35(1):13–32, 1999.
29. H. Sundar, D. Silver, N. Gagvani, and S. Dickinson. Skeleton Based Shape Matching and Retrieval. In *International Conference On Shape Modeling International and Applications*, pages 130–142, Seoul, Korea, May 2003.
30. D. Vranic and D. Saupe. 3-D Model Retrieval With Spherical Harmonics and Moments. In *Proceedings of the DAGM*, pages 392–397, 2001.