

# From Skeletons to Bone Graphs: Medial Abstraction for Object Recognition

Diego Macrini  
University of Toronto

Kaleem Siddiqi  
McGill University

Sven Dickinson  
University of Toronto

## Abstract

*Medial descriptions, such as shock graphs, have gained significant momentum in the shape-based object recognition community due to their invariance to translation, rotation, scale and articulation and their ability to cope with moderate amounts of within-class deformation. While they attempt to decompose a shape into a set of parts, this decomposition can suffer from ligature-induced instability. In particular, the addition of even a small part can have a dramatic impact on the representation in the vicinity of its attachment. We present an algorithm for identifying and representing the ligature structure, and restoring the non-ligature structures that remain. This leads to a **bone graph**, a new medial shape abstraction that captures a more intuitive notion of an object's parts than a skeleton or a shock graph, and offers improved stability and within-class deformation invariance. We demonstrate these advantages by comparing the use of bone graphs to shock graphs in a set of view-based object recognition and pose estimation trials.*

## 1. Introduction

The recognition of shape categories requires a parts-based representation which is invariant to image translation, rotation, and scale, as well as part articulation and within-class shape deformation. In 3-D, Binford's generalized cylinders [4] decomposed an object into a set of elongated parts defined by sweeping a 2-D cross section through a 3-D space curve. The concept of an axial description of shape was proposed even earlier in 2-D through Blum's medial axis transform, or skeleton [5]. Although skeletal descriptions have been a mainstay in the pattern recognition community for many years, the computer vision community has focused largely on appearance-based representations for recognition. However, as the community evolves from appearance-based exemplar recognition to shape-based category recognition, skeleton-based descriptions have again emerged as a powerful representation [15, 13].

Skeletonization algorithms map a closed 2-D shape into a set of medial branches that terminate at endpoints or

branch junctions. An important criterion for any effective shape representation is locality of representation, *i.e.*, the impact of a local boundary perturbation should be local. Unfortunately, shape skeletons can fail to meet this criterion. A key issue is that skeletal branches do not always map, in a one-to-one fashion, to intuitive shape parts. As one example, the shorter rear leg of the dog in Fig. 1(a) results in an incident branch that oversegments the representation of the dog's body into two skeletal branches. A zoom-in on the junction also reveals a local cusp on the body branch, which arises due to the attachment. A similar situation arises near the two front legs. The representation of each of the four legs, as well as the tail, is also undersegmented in the sense that the associated skeletal branches extend well past the locations of the part attachments with the body. It is important to emphasize that these problems of branch over- and undersegmentation are orthogonal to the problem of handling "noise" due to boundary perturbations. Whereas a number of effective pruning techniques have been proposed for the latter [14, 17, 2], they do not address the issue that the topology of the skeleton can do a poor job of reflecting the salient part structure of the shape.

One alternative to mitigating the sensitivity of the topological structure of the skeleton is to pass this instability to the matching algorithm. The fact that two very similar shapes may yield skeletons which are topologically dissimilar means that the matching algorithm must be able to somehow search for correspondences at higher levels of abstraction, *i.e.*, levels above the structural instability. A number of elegant graph matching frameworks have evolved to address this very challenging problem [15, 13]. In fact, in [13] the transitions of the medial axis [10] are themselves incorporated in edit-distance operations which allow topologically distinct skeletal structures due to similar shapes to be matched.

In the current paper we argue for a novel representation which can improve the performance of all skeletal matching algorithms. We use *ligature* analysis [5] to devise a more stable skeletal representation, where skeletal segments map in a one-to-one fashion to intuitive shape parts. Intuitively, ligature regions (shown as green curves in Fig. 1) are segments of the skeleton that contribute little to the reconstruct-

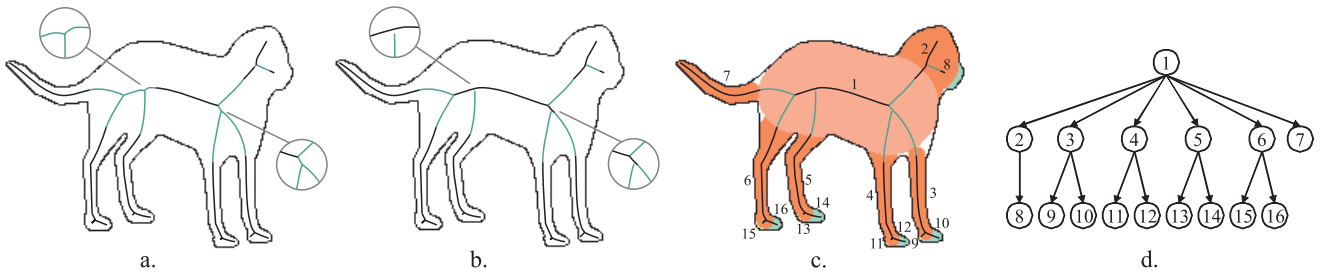


Figure 1. Intuitive Part Decomposition: (a) skeletal instability arises from part oversegmentation and undersegmentation. For example, the skeletal branch associated with the dog’s body is oversegmented by the skeletal segment extending from the shorter rear leg. A similar situation occurs near the front legs. The vicinity of the oversegmentation is enlarged in each case, showing the resulting perturbation of the skeleton. Those skeletal segments shown in green are ligature regions, and they contribute little to the shape of the object. A purely local analysis of ligature is problematic in the presence of such oversegmentation, as illustrated by the non-intuitive labeling of the body part in the vicinity of the oversegmentation as ligature. (b) Our algorithm for detecting and removing ligature-induced skeletal instability uses a novel local ligature analysis to first identify and rectify skeletal branch oversegmentation. (c) A second ligature analysis then yields a set of salient parts, called *bones* (shown in black). The bones capture the coarse part structure of the object, as indicated by the colored parts reconstructed from the bones. (d) The bones give rise to a *bone graph*, an intuitive and stable representation whose nodes represent the salient parts and whose edges, derived from the final ligature analysis, capture part attachment.

tion and representation of the boundary. One could simply remove these portions of the skeleton, as suggested in [1]. However, not every ligature point is a good candidate for removal, as illustrated in Fig. 1(a), where much of the dog’s body skeleton (toward its back legs) is labeled as ligature. Clearly, the oversegmentation problem cannot be separated from the ligature analysis.

In a first stage of ligature analysis we identify the cases of branch oversegmentation, and proceed to rectify the oversegmentation through a branch fusion and restoration process. This yields a set of branches that adhere to the geometric properties of the medial axis. Once the oversegmentation is corrected, a second ligature analysis yields a new set of ligature segments arising from branch undersegmentation. Fig. 1(b) shows the final ligature analysis, following the correction of branch oversegmentation. As can be seen in the zoom-ins on the junctions, the two cases of branch oversegmentation have been identified and corrected. Moreover, the second ligature analysis has effectively extended the (non-ligature, shown in black) representation of the dog’s body.

The final set of non-ligature branches captures the salient medial parts of the object. In fact, they yield a reconstructed shape that is very similar to that of the original object, as shown in Fig. 1 (c) by the union of the colored regions, each of which is the reconstruction of one non-ligature (black) branch. We assemble the restored non-ligature segments in a novel, parts-based shape representation called the *bone graph*, whose nodes represent stable, intuitive skeletal parts (*bones*), and whose attachment edges are derived from the ligature segments. To demonstrate the improved stability of the bone graph, we compare it head-to-head with the ubiquitous shock graph [15, 13] in a set of view-based 3-D object

recognition and pose estimation trials. Experimental evidence demonstrates that the bone graph is far more invariant to viewpoint change-induced perturbations to a silhouette than the shock graph, leading to significantly improved performance.

## 2. Ligature and its Geometry

The notion of ligature as a type of “glue” between parts was first proposed by Blum and Nagel [6]. Much later, it was revisited in [1] with several qualitative examples provided to demonstrate that with ligature regions labeled, the remaining portions of the skeleton appeared to be stable. In fact, the development in [1] was guided by a principle of ligature removal: reconstructions without ligature appeared to lose little in terms of boundary detail. It was also shown that certain deformations of the boundary, such as those resulting from evolution by curvature [11] could swiftly lead to topological changes in ligature regions, a further motivation for removal. However, these developments fell short of a precise computational method by which ligature analysis could lead to a higher level of abstraction. Furthermore, no direct attempt was made to apply these ideas to the then emerging techniques for skeletal graph matching, apart from the notion that ligature regions should be viewed as less salient.

Let  $\Omega$  be the set of all points  $(x, y)$  within the interior of a 2-D object, with  $\mathcal{S}(\Omega)$  its medial axis [5]. Each skeletal point in  $\mathcal{S}$  is characterized by a position  $\mathbf{p}$  and a radius  $r$ . The relationship between the object angle  $\theta$ , the spoke vectors  $b^{\pm 1}$ , and the direction of the unit tangent vector  $\mathbf{T}$  (see

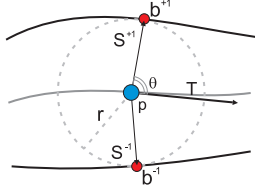


Figure 2. Local geometry of a skeletal curve. The maximum inscribed disc at a regular skeletal point  $\mathbf{p}$  with radius  $r$ , touches the boundary at two bitangents points  $b^{+1}$  and  $b^{-1}$ , defining two spokes emanating from  $\mathbf{p}$ . The angle between the unit tangent  $\mathbf{T}$  (to the medial curve) and either spoke is  $\theta$ , the *object angle*.

Fig. 2) is given by

$$\theta = \arccos\left(-\frac{dr}{d\tilde{s}}\right), \quad (1)$$

where  $\tilde{s}$  is the arc length along the medial curve. The object angle is expressed with respect to the unit tangent in the direction of decreasing radius along the curve. A **full**-ligature segment is the set of connected points of a skeletal branch associated with a pair of opposing concave corners on the shape’s boundary. Similarly, a **semi**-ligature segment is the set of connected points of a skeletal branch associated with a single concave corner on the shape’s boundary (to one side of the segment).

The taxonomy of generic full- and semi-ligature cases is shown in Fig. 3, with full-ligature segments shown in red and semi-ligature segments shown in blue (note that both full- and semi-ligature segments were colored green in Fig. 1). We make a distinction between the ligature segments that connect two shape parts in an “end to side” configuration (Fig. 3 (a) and (d)), and the ligature segments that connect two shape parts in an “end to end” configuration (Fig. 3 (b) and (c)). For the end-to-side attachment case in Fig. 3 (a), the blue cusp actually represents two small semi-ligature segments which meet at the apex of the cusp. Ligature configurations can be quite complex, particularly when they are nested. The cases of full-on-full, full-on-semi and full-on-full (branch-crossing) configurations are illustrated in Figs. 3 (d), (e) and (f), respectively.

### 3. Detecting Ligature

The algorithm for ligature detection developed in [1] works by first identifying strong negative curvature minima on the boundary (by using a curvature measure) and then labeling those skeletal points whose bitangent points  $b^{+1}, b^{-1}$  fall within an  $\epsilon$ -ball of the same negative curvature minima. In this approach the negative curvature minima are computed at a fixed (boundary length) scale. We seek a more robust approach in which local shape properties dictate the scale of negative curvature minima detection.

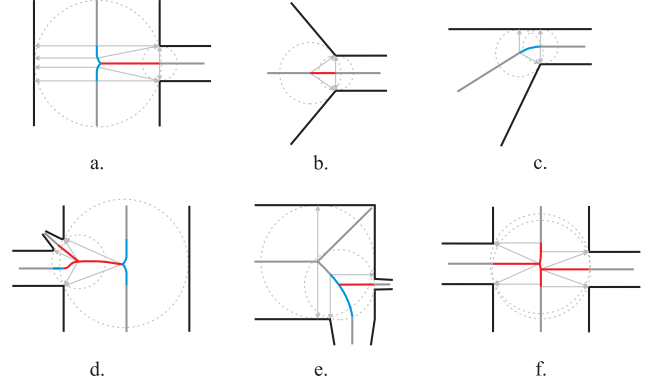


Figure 3. Types of ligature segment configurations (the basic cases a, b, and c are adapted from [1]): (a) Full ligature (red) and semi-ligature (blue) segments (induced by a pair of concave corners) representing an “end to side” part attachment – the blue cusp actually represents two adjacent semi-ligature segments that meet at the apex of the cusp; (b) Full ligature (red) segment (induced by a pair of concave corners) representing an “end to end” part attachment; (c) Semi-ligature (blue) segment (induced by a single concave corner) representing an “end to end” part attachment; (d) and (e) Examples of complex configurations of full-on-full and full-on-semi nested ligature. (f) Example of a complex full-on-full configuration, representing a potential “crossing” of branches.

To this end, we combine a stable measure of boundary-to-axis ratio (BAR) [6] with the detection of negative curvature minima at the appropriate scale (Fig. 4). In particular, we detect high curvature points by following the method presented in [7].

**Definition 1.** For any regular skeletal segment, there are two **boundary-to-axis ratios**, each obtained by dividing the length of the associated boundary on each side by the length of the segment. The boundary segment is defined by the spoke endpoints corresponding to the first and last point of the skeletal segment.

**Definition 2.** A **ligature** segment has at least one side whose boundary-to-axis ratio is smaller than one, and which defines a boundary interval of negative curvature points that corresponds to a curvature minimum at a scale in which the length of the boundary interval is zero.

We use the above definition to find all maximal ligature segments with monotonically varying radii. The radii at the endpoints of these segments are used in Section 4 to determine the oversegmented branches that must be fused together. Ligature detection is initially applied to all branches in the input skeleton, and subsequently to every restored branch produced by the fusion operation described in the next section.

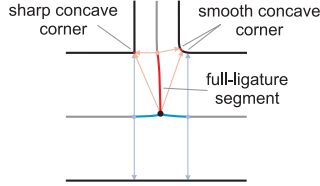


Figure 4. Ligature from concave corners detected at different scales. A sharp concave corner (left) has the property that the spoke vectors associated with the ligature segment coincide at it. In contrast, a smooth concave corner (right) is traced out by the endpoints of non-coincidental spoke vectors. In both situations the boundary-to-axis ratio is less than one.

#### 4. Restoring Over- and Undersegmented Parts

In order to address the problem that more than one skeletal segment can represent the same shape part we propose to merge the appropriate branch pieces. Intuitively, we shall analyze ligature properties around every branch junction in the medial axis and associate locations of oversegmentation with *end-to-side* (ES) attachments between branches. The remaining branch junctions will correspond to *end-to-end* (EE) attachments, *i.e.*, branches that are connected at their endpoints. For an ES attachment the identification of the pair of branches that should be grouped requires the use of appropriate criteria, such as the relative thickness of attached parts and/or the continuity of skeletal branches.

**Definition 3.** For any end-to-side attachment, there is a junction point with three or more branches incident on it, two of which map to the same shape part. This pair of branches is said to be **broken**, while the other incident branches are said to be **attached**.

Without loss of generality, we focus our analysis on generic junction points. A junction is said to be generic if the degree of incident branches under arbitrarily small perturbations of the boundary curve is maintained [10]. In this case, there are two types of ES attachments, which we refer to as *simple* and *complex*. A simple attachment has its ligature segments induced by one or two concave corners, while a complex attachment is related to three or more concave corners.

It can be shown that simple attachments with one concave corner induce two semi-ligature segments (Fig. 5 (a)), while those with two corners induce one full-ligature segment and two semi-ligature segments (Fig. 4). Unfortunately, this ligature information alone is generally not enough to distinguish between ES and EE attachments. We resolve this ambiguity by interpreting protrusions that are smaller than the object width at the junction point as ES attachments. Specifically, a junction with simple ligature properties is classified as an ES attachment if it defines exactly one boundary gap, with the distance between the end-

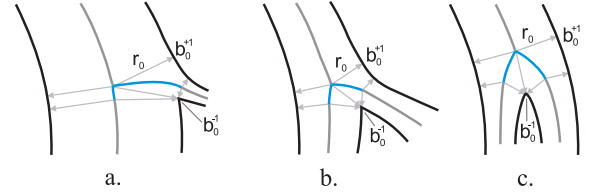


Figure 5. Example of ES and EE attachments with similar ligature properties. In each example there is a junction point associated with one concave corner,  $b_0^{-1}$ , which induces two semi-ligature segments (blue points). In order to classify the junction, the length of the boundary gap defined by the spoke endpoints  $b_0^{+1}$  and  $b_0^{-1}$  is compared against the radius of the junction,  $r_0$ . The condition that  $\|b_0^{+1} - b_0^{-1}\|_2 < r_0$  is only met by case (a), which is classified as an ES attachment. The junctions in (b) and (c) are classified as EE attachments.

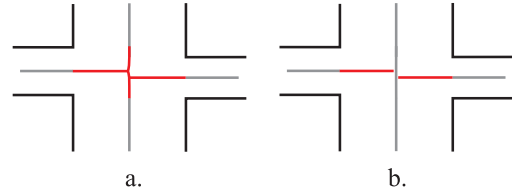


Figure 6. (a) A crossing is formed by two junctions connected by ligature segments. The labeling of broken and attached branches in this configuration defines different part decompositions. (b) An interpretation of the crossing as a vertical part with two horizontal end-to-side parts attached to it.

points of the spokes  $b_0^{+1}$  and  $b_0^{-1}$  (see Fig. 5) being smaller than the radius  $r_0$  at the junction point, and as an EE attachment otherwise.

Complex attachments correspond to cases of either *nested ligature* or *branch crossings*. A nested-ligature configuration arises when there are at least two attachment points and one branch that is both broken and attached. This defines a parent-child relationship between junction points in which the boundary gap defined by the spokes of the child junction is included within that defined by the parent junction (*e.g.*, Fig. 3 (d) and Fig. 8).

The set of complex ES attachments that are not cases of nested ligature correspond to branch crossings (Fig. 6) where two ES attachments are connected by ligature points, with their attached branches on opposing sides of the broken branches. We take the approach of interpreting each such crossing as one broken branch with two ES attachments, as shown in Fig. 6 (b). We compare the radii of all the ligature segments incident to each junction and label as broken the pairs of branches with the largest radius values. There are other possible interpretations of a crossing, the consideration of which depends on perceptual preferences which may be domain dependent. This is left for future work.

With the above classification based on a ligature analy-

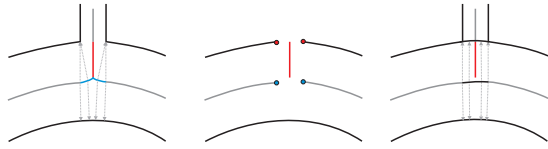


Figure 7. The branch fusion operation. LEFT: A simple ES attachment with two semi-ligature “arms”. The positions and radii of the semi-ligature points are “corrupted” by the opening along the boundary formed by the protrusion. MIDDLE: The positions and radii of the skeletal gap endpoints (blue circles) are interpolated in order to recreate the medial axis of the shape part without the protrusion; RIGHT: The skeletal gap is now interpolated with the spokes computed using the relation between the object angle and radius value given by Eq. 1.

sis at junction points, we now complete the restoration of oversegmented parts. We begin by evaluating the cases of complex, nested attachments which involve pairs of junctions connected by ligature segments. In the case of nested attachments, we fuse the broken branches of the child attachment before that of the parent attachment, such that the nestedness is reduced to two simpler cases. This step is applied recursively until the basic case of a simple attachment is reached. A junction point that is not a complex or simple ES attachment is classified as an EE attachment.

The process of fusing the two broken branches of a ES attachment is an attempt to recreate the medial axis that a shape part would have if the protruding part were removed, and the boundary opening formed by it were filled with a smooth curve (see Fig. 7). The skeletal gap is replaced by a smooth interpolation of the positions and radius values of its endpoints while respecting tangent continuity. The boundary gap is then filled by calculating the spokes of the interpolated points [9], as specified by Eq. 1 (Fig. 7 (right)). In our implementation, we perform a cubic spline interpolation of the gap’s medial curve, and a linear interpolation of the gap’s radius function.

The fusion of branches, together with the restoration of medial axis properties, plays a significant role. The restoration of the boundary gap is necessary to compute the BAR of the parent branch in nested ligature cases (Fig. 8) and the restoration of skeletal information simplifies the comparison between similar shapes with missing parts (the individual parts now encode a similar boundary contour). However, the skeleton can become disconnected as a result of each fusion operation, since the original branch junction points are not necessarily interpolated (*e.g.*, see the red and black points in Fig. 7 (right)). While it is relatively simple to reconnect the skeleton by extending the endpoint of the disconnected branch until it intercepts the fused medial axis, we choose instead to preserve the original branch connectivity by maintaining a set of adjacency relations produced by branch fusions. In particular, we assume that the connection

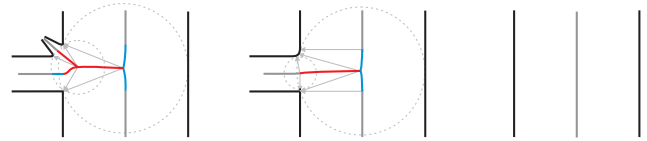


Figure 8. Example of branch fusions helping in the restoration of nested ligature. LEFT: Nested boundary gaps form nested ligature; MIDDLE: The child ES attachment was restored first, the inner boundary gap is now filled, and the ligature points are recomputed; RIGHT: The parent ES attachment becomes a *simple* case (Sec. 4) and is restored accordingly.

point on the fused branch corresponds to the skeletal point closest to the original junction point.

The final stage of the ligature-based restoration process involves the decomposition of undersegmented skeletal branches. Specifically, we partition each branch into ligature and non-ligature segments. The resulting non-ligature segments play the role of salient parts or **bones** and the ligature segments act as the “glue” to hold them together.

## 5. Constructing The Bone Graph

The second major component of our framework takes a restored skeleton and constructs its bone graph. In this representation, the salient parts (the bones) are mapped to nodes, while the identified ligature segments and the remaining branch junctions are mapped to edges in the graph. The attributes of nodes and edges are the same, *i.e.*, both contain a sequence of skeletal points. In the construction of the graph, we seek an encoding of the attachment types, and the elimination of the uninformative complexity of ligature-on-ligature attachments. Furthermore, inspired by the shock graph, we also seek a (partial) hierarchical relation between nodes.

We eliminate the uninformative ligature-on-ligature structures by transforming them into bone attachments. These structures correspond to tertiary or higher order relations between bones, but can be transformed into a set of binary bone relations by selecting one of the bones as the parent of the others. To this end, we define the parent bone as the bone connected to the ligature point with largest radius value (Fig. 9). The attributes of each parent-child edge are given by the skeletal points of the ligature segment that is immediately adjacent to each child bone. As a result, every ligature point is uniquely associated with one edge, and every edge connects two bones.

Like the shock graph, we direct the edges of the bone graph according to a local estimate of relative part size. In addition, we allow for undirected edges that represent the uncertainty in the part hierarchy that arises at a “neck” shape (Fig. 9,  $l_3$  and  $l_4$ ). Let  $\mathcal{A}$  be the set of directed edges, and  $\mathcal{E}$  be the set of undirected edges. A directed bone rela-

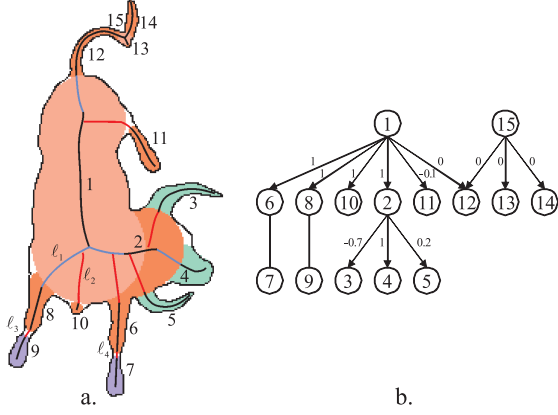


Figure 9. Example of a bone graph. (a) The ligature-on-ligature structures of the restored skeleton, such as that formed by the ligature segments  $\ell_1$  and  $\ell_2$ , are expressed in (b) as edges between the parent node 1 and the two child nodes 8 and 10. The shape areas associated with each bone are colored differently for each level of the graph, and drawn following the edge directions in bottom-up order. The undirected edges correspond to ligature points with non-decreasing radius functions formed by neck shapes, such as  $\ell_3$  and  $\ell_4$ . See text for explanation of edge labels.

tion  $(i, j) \in \mathcal{A}$ , directed from bone  $i$  to bone  $j$ , reflects one of the following conditions:

- bones  $i$  and  $j$  are incident to a junction point, and the radius function of bone  $i$  is constant or increases away from the junction. This is the case where a larger bone branches out to form a series of smaller bones. The attribute of such an edge is the empty sequence.
- bones  $i$  and  $j$  share a junction point at which the radius function is a local maximum. In this case, bone  $i$  in fact has the junction point as its only skeletal point. The attribute of such an edge is the empty sequence.
- bones  $i$  and  $j$  are connected by ligature points whose radius function decreases monotonically from  $i$  to  $j$ . This is the case where there is an end-to-side attachment between bones  $j$  and  $i$ , or where the bones are connected end-to-end by one single ligature segment or by nested ligature.

An undirected bone relation  $(i, j) \in \mathcal{E}$  indicates that bone  $i$  is connected end-to-end by a ligature segment that has a non-decreasing radius function to bone  $j$ .

For ES attachments, we encode the position,  $p_{i,j}$ , along the side of parent bone  $i$  of the attachment, of the end of the child bone  $j$ . For convenience, we normalize the length of each bone’s medial curve to the interval  $[0, 1]$ , with the “0” end chosen arbitrarily. Assuming a clockwise traversal of the branch from the “0” end, attachments on the left side are specified as positive values in the open interval  $(0, 1)$  (a

value of 0 or 1 would imply an EE attachment), while attachments on the right side are specified as negative values in the open interval  $(0, -1)$ .<sup>1</sup> Such an attachment specification allows us to qualitatively distinguish whether ES attachments are near the ends or the middle of a bone, whether multiple attachments are on the same or opposite sides of a bone, or whether the attachments on the same side of a bone are near or far apart.

For directed EE attachments, we encode the position,  $p_{i,j}$ , at the end of parent bone  $i$  of the attachment of the end of child bone  $j$  as 0 or 1. While this is again ambiguous due to choice of endpoint, it does allow us to quickly distinguish whether bones are attached to the same or opposite ends. Finally, since the undirected EE attachments occur only at necks, their attachment structure is fixed and an additional attachment specification would be extraneous. The ability to facilitate such qualitative attachment judgements is inspired by Biederman’s RBC theory [3]. We can now formally define the bone graph:

**Definition 4.** The Bone Graph of a 2-D shape,  $\text{BG}(\Omega)$ , is an edge labeled mixed graph  $G = (V, E, A, \gamma)$  with

- **vertices (bones)**  $V = \{1, \dots, n\}$ , representing the non-ligature segments of the restored skeleton;
- **undirected edges**  $(i, j) \in E \subseteq V \times V$  iff  $(i, j) \in \mathcal{E}$ ;
- **directed edges**  $(i, j) \in A \subseteq V \times V$  directed from vertex  $i$  to vertex  $j$  iff  $(i, j) \in \mathcal{A}$ ; and
- **directed edge labels**  $\gamma : A \mapsto [-1, 1]$ , with  $\gamma(i, j) = p_{i,j}$ .

Note that node and edge attributes also encode the position, radius, tangent, and object angle of each skeletal point associated with them. This information is used at matching time to compare the geometrical properties of nodes and edges, and can also be used to reconstruct the original shape.

## 6. Experiments

We evaluate the structural stability of the bone graph by comparing it against the shock graph in a set of view-based object recognition experiments. We provide a meaningful comparison by evaluating both types of graphs under the same graph matching framework and by using the same node similarity function. We follow the matching framework of [15], and construct a node similarity function for bone graphs by sub-partitioning each bone into shock parts. While this matching framework ignores the edge attributes and labels of the bone graph, as well as the undirected edges, it does allow us to directly compare the stability of

<sup>1</sup>In fact, there are two possible attachment specifications, depending on the choice of endpoint, and both have to be considered if edge label signs are used in matching or other tasks.

these two medial descriptions by ensuring that nodes and edges are interpreted identically.

We begin with a database of 1664 silhouette views of 13 3-D models (128 uniformly sampled views per object), yielding a database of shock graphs and a database of bone graphs<sup>2</sup>. Each view is successively removed and compared to each remaining view. If the 3-D model from which the closest matching view was generated is the same as that of the query, then recognition (identification) is said to be successful. If recognition is successful *and* the closest matching view is an immediate neighbor (on the viewsphere) of the removed view, then pose estimation is said to be successful. The fact that the silhouettes represent views of 13 objects means that many of the silhouettes are very similar, thereby dramatically increasing the ambiguity in the database for the pose estimation trials (compared to 1664 silhouettes of 1664 distinct objects).

In the next set of trials, each of the 1664 views is again used as a query. However, the database of views is subsampled by randomly removing 25% of the views, leading to subsampled databases of shock graphs and bone graphs. The same experiment is repeated, measuring correct recognition rates for shock graphs and bone graphs. This subsampling/evaluation process is repeated down to, and including, databases containing only 32 views of each object (75% model view removal). At each iteration, we compute three separate random viewsphere subsamplings and aggregate the results. In this fashion 16,640 recognition trials are conducted in total.

Figure 10 plots both the recognition and pose estimation success rates for both shock graphs and bone graphs as a function of decreasing viewsphere sampling resolution. For the more tolerant recognition task, the improved stability of the bone graph over the shock graph is clearly visible. The results show an improvement of approximately 3% with no model views removed, and this improvement increases steadily to approximately 7% with 75% of the model views removed. The pose estimation results, reflecting a far more stringent recognition task, show a dramatic (13%) improvement in stability over the shock graph at all sampling resolutions. We remind the reader that these experiments do not exploit the full power of the bone graph in that the relative location of attachments (bone graph directed edge labels) are ignored so as to put the bone graph on the same footing as the shock graph for each trial. Exploiting such constraints in the matcher would undoubtedly lead to further improvement in the results.

Figure 11 illustrates a number of successful matches drawn from the experiment. In each pair, the shape on top represents the query while the shape underneath represents the closest matching database view. For both shapes, the

<sup>2</sup>We use the publicly available shape database and shock graph software from ([www.cs.toronto.edu/~dmac/ShapeMatcher/](http://www.cs.toronto.edu/~dmac/ShapeMatcher/)).

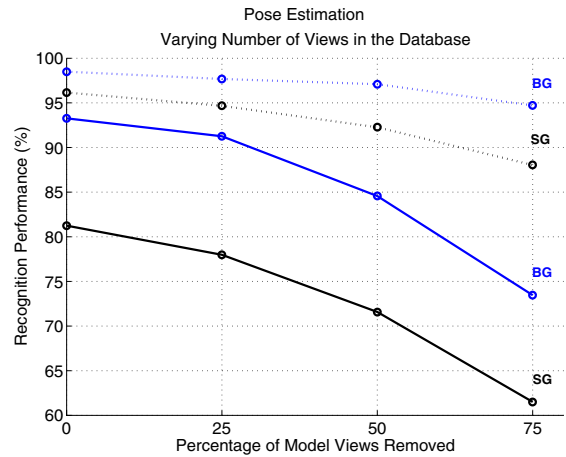


Figure 10. Correct recognition (dashed curves) and pose estimation (solid curves) rates for bone graphs (blue, labeled BG) and shock graphs (black, labeled SG) as a function of decreasing viewsphere sampling resolution. The bone graph clearly exhibits superior stability for both tasks, with dramatic improvement for the more stringent pose estimation task.

recovered bones are displayed (shaded) over the restored skeletons, with the final ligature/non-ligature analysis reflected in the coloring of the skeletons. In addition, corresponding bones between query and model, as computed by the matcher, are colored the same. These examples illustrate the fact that while viewpoint changes may induce significant structural variation in the skeleton, due to skeleton over- and under-segmentation, the final bone decomposition is far more stable and viewpoint invariant. Whereas the shock graph is forced to explicitly encode this structural instability, the bone graph captures the salient shape at a higher and far more stable level of abstraction.

## 7. Conclusions

The instability of the skeleton due to part attachment has long been acknowledged by the shape representation community. While analysis of this instability has yielded stability measures ranging from very local (skeleton point) [12] to semi-local (skeleton branches) [17], most efforts can be viewed as *skeleton processing*, mapping input skeletons to output skeletons rather than mapping input skeletons to higher level shape abstractions. Moreover, evaluation is typically anecdotal (visual), lacking the context of a particular (e.g., recognition) task. The one exception is the shock graph [15, 13], whose one-to-many mapping of skeletal segments to abstract parts unfortunately carries forward this skeletal instability.

In this article we introduce a novel abstraction based on the skeleton where the goal is to map skeletal segments to intuitive shape parts. We do so by first extending the def-

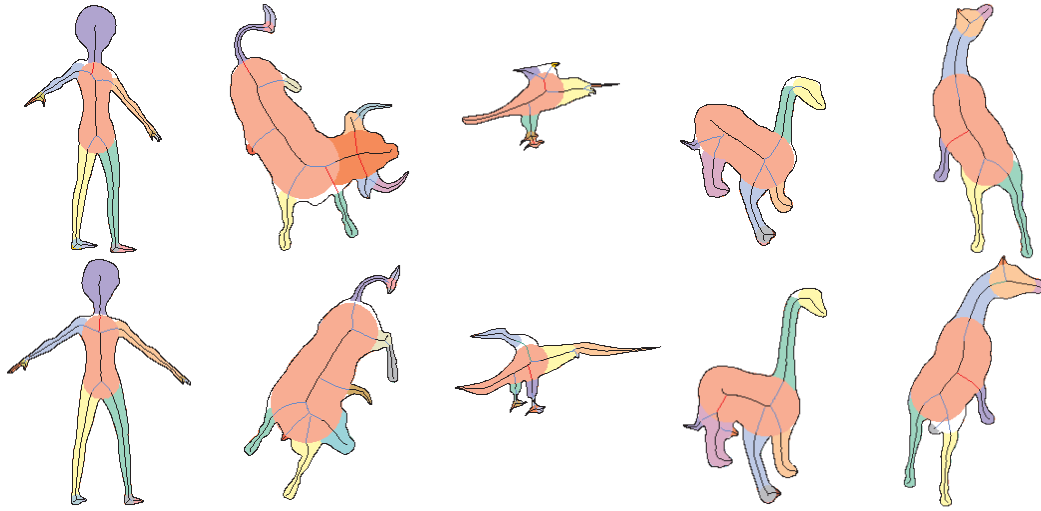


Figure 11. Matching Bone Graphs. In each pair of shapes, the top shape represents the query while the bottom shape represents the closest matching database view. Each shape includes its final restored skeleton, along with the shaded bones defined by the non-ligature segments. Corresponding bones between query and model, as computed by the matcher (which ignores part order), are colored the same. Close examination reveals that while skeleton topology (encoded explicitly in a shock graph) may change significantly due to changes in viewpoint, bone graph topology is far more stable.

inition of ligature [1] to account for smooth concave corners while incorporating the appropriate scale information for their detection along the shape boundary. This improved ligature analysis is then used to restore over- and undersegmented parts, leading to a new skeleton. Whereas related skeletal segmentation methods stop at this stage, including [17, 16, 12, 8], we go one step further by introducing a novel graph abstraction of medial structure that partitions the restored skeletal branches into intuitive parts. In this fashion a set of meaningful parts and their attachment relations are recovered from ligature structure. The resulting bone graph offers a powerful parts-based shape abstraction whose stability is demonstrably better than the shock graph in a view based object recognition and pose estimation task. In future work we plan to develop a matcher that exploits the edge attribute labels in this representation (reflecting “where” parts are attached) and further to explore the bone graph as a tool for viewsphere partitioning.

## References

- [1] J. August, K. Siddiqi, and S. Zucker. Ligature instabilities in the perceptual organization of shape. *CVIU*, 76(3):231–243, 1999.
- [2] X. Bai, L. J. Latecki, and W.-Y. Liu. Skeleton pruning by contour partitioning with discrete curve evolution. *PAMI*, 29(3):449–462, March 2007.
- [3] I. Biederman. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.
- [4] T. Binford. Visual perception by computer. In *IEEE Conference on Systems and Control*, Miami, FL, 1971.
- [5] H. Blum. Biological shape and visual science. *Journal of Theoretical Biology*, 38:205–287, 1973.
- [6] H. Blum and R. N. Nagel. Shape description using weighted symmetric axis features. *Patt. Recogn.*, 10:167–180, 1978.
- [7] D. Chetverikov and Z. Szabo. A simple and efficient algorithm for detection of high curvature points in planar curves. In *Workshop of the Austrian Pattern Recognition Group*, pages 175–184, 1999.
- [8] J. Feldman and M. Singh. Bayesian estimation of the shape skeleton. *PNAS*, 103:18014–18019, 2006.
- [9] P. J. Giblin and B. B. Kimia. On the intrinsic reconstruction of shape from its symmetries. *PAMI*, 25(7):895–911, 2003.
- [10] P. J. Giblin and B. B. Kimia. On the local form and transitions of symmetry sets, medial axes, and shocks. *IJCV*, 54(1-3):143–156, 2003.
- [11] M. A. Grayson. Shortening embedded curves. *The Annals of Mathematics*, 129:71–111, 1989.
- [12] R. A. Katz and S. M. Pizer. Untangling the Blum medial axis transform. *IJCV*, 55(2-3):139–153, 2003.
- [13] T. Sebastian, P. N. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *PAMI*, 26(5):550–571, 2004.
- [14] D. Shaked and A. M. Bruckstein. Pruning medial axes. *CVIU*, 69(2):156–169, 1998.
- [15] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *IJCV*, 35(1):13–32, 1999.
- [16] S. Svensson and G. S. di Baja. Using distance transforms to decompose 3d discrete objects. *IVC*, 20(8):529–540, 2002.
- [17] H. Tek and B. B. Kimia. Boundary smoothing via symmetry transforms. *JMIV*, 14(3):211–223, 2001.