Multiscale Symmetric Part Detection and Grouping

Alex Levinshtein, Sven Dickinson
University of Toronto
babalex,sven@cs.toronto.edu

Cristian Sminchisescu
University of Bonn
cristian.sminchisescu@ins.uni-bonn.de

Abstract

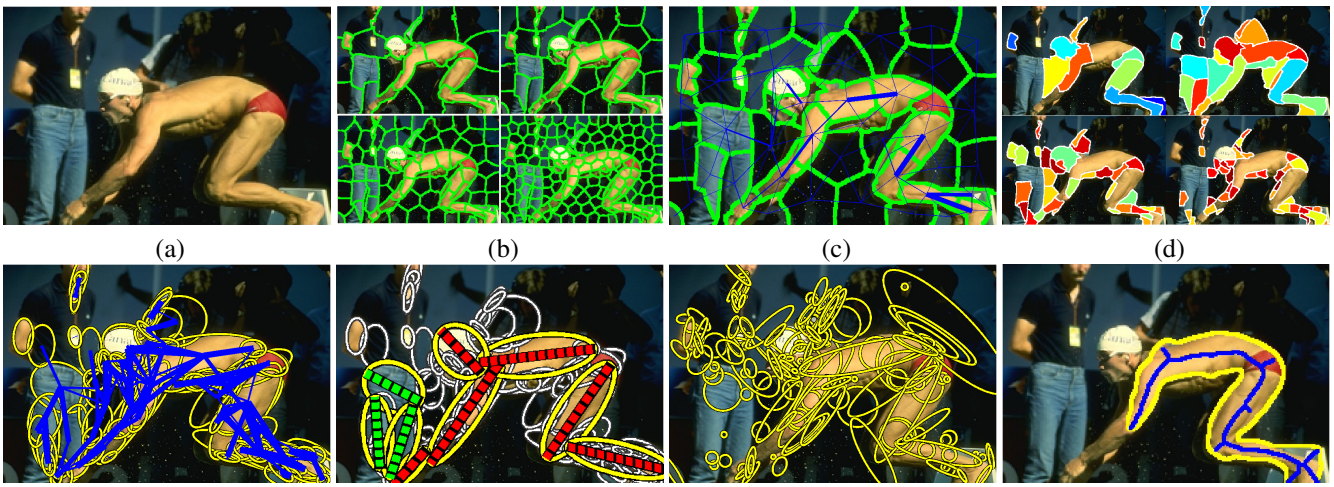
Skeletonization algorithms typically decompose an object's silhouette into a set of symmetric parts, offering a powerful representation for shape categorization. However, having access to an object's silhouette assumes correct figure-ground segmentation, leading to a disconnect with the mainstream categorization community, which attempts to recognize objects from cluttered images. In this paper, we present a novel approach to recovering and grouping the symmetric parts of an object from a cluttered scene. We begin by using a multiresolution superpixel segmentation to generate medial point hypotheses, and use a learned affinity function to perceptually group nearby medial points likely to belong to the same medial branch. In the next stage, we learn higher granularity affinity functions to group the resulting medial branches likely to belong to the same object. The resulting framework yields a skeletal approximation that's free of many of the instabilities plaguing traditional skeletons. More importantly, it doesn't require a closed contour, enabling the application of skeleton-based categorization systems to more realistic imagery.

1. Introduction

The medial axis transform [3] decomposes a closed 2-D shape into a set of skeletal parts and their connections, providing a powerful parts-based decomposition of the shape that's suitable for shape matching [19, 16]. While the medial axis-based research community is both active and diverse, it has not kept pace with the mainstream object recognition (categorization) community that seeks to recognize objects from cluttered scenes. The main reason for this disconnect is the restrictive assumption that the silhouette of an object is available – that the open problem of figure-ground segmentation has somehow been solved. Even if it were possible to segment the figure from the ground, a second source of concern arises around the instability of the resulting skeleton – the skeletal branches often don't map one-to-one to the object's coarse symmetric parts. However, these limitations should in no way deter us from the goal of recovering an object's symmetric part structure from images. We simply need an alternative approach that doesn't assume figure-ground segmentation and doesn't introduce skeletal instability.

In this paper, we introduce a novel approach to recovering the symmetric part structure of an object from a cluttered image, as outlined in Fig. 1. Drawing on the principle that a skeleton is defined as the locus of *medial points*, i.e., centers of maximally inscribed disks, we first hypothesize a sparse set of medial points at multiple scales by segmenting the image (Fig. 1(a)) into compact superpixels at different superpixel resolutions (Fig. 1(b)). Superpixels are adequate for this task, balancing a data-driven component that's attracted to shape boundaries while maintaining a high degree of compactness. The superpixels (medial point hypotheses) at each scale are linked into a graph, with edges adjoining adjacent superpixels. Each edge is assigned an affinity that reflects the degree to which two adjacent superpixels represent medial points belonging to the same symmetric part (medial branch) (Fig. 1(c)). The affinities are learned from a set of training images whose symmetric parts have been manually identified. A standard graph-based segmentation algorithm applied to each scale yields a set of superpixel clusters which, in turn, yield a set of regularized symmetric parts (Fig. 1(d)).

In the second phase of our approach, we address the problem of perceptually grouping symmetric parts arising in the first phase. Like in any grouping problem, our goal is to identify sets of parts that are causally related, i.e., unlikely to co-occur by accident. Again, we adopt a graph-based approach in which the set of symmetric parts across all scales are connected in a graph, with edges adjoining parts in close spatial proximity (Fig. 1(e)). Each edge is assigned an affinity, this time reflecting the degree to which two nearby parts are believed to be physically attached. Like in the first phase, the associated, higher granularity affinities are learned from the regularities of attached symmetric parts identified in training data, and the same graph-based segmentation algorithm is applied to yield part clusters, each representing a set of regularized symmetric elements and their hypothesized attachments (Fig. 1(f)).



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 1. Overview of our approach for multiscale symmetric part detection and grouping: (a) original image; (b) set of multiscale superpixel segmentations (different superpixel resolutions); (c) the graph of affinities shown for one scale (superpixel resolution); (d) the set of regularized symmetric parts extracted from all scales through a standard graph-based segmentation algorithm; (e) the graph of affinities between nearby symmetric parts (all scales); (f) the most prominent part clusters extracted from a standard graph-based segmentation algorithm, with abstracted symmetry axes overlaid onto the abstracted parts; (g) in contrast, a Laplacian-based multiscale blob and ridge decomposition, such as that computed by [10], shown, yields many false positive and false negative parts; (h) in contrast, classical skeletonization algorithms require a closed contour which, for real images, must be approximated by a region boundary. In this case, the parameters of the N-cuts algorithm [17] were tuned to give the best region (maximal size without region undersegmentation) for the swimmer. A standard medial axis extraction algorithm applied to the smoothed silhouette produces a skeleton (shown in blue) that contains spurious branches, branch instability, and poor part delineation.

Our approach offers clear advantages over competing approaches. For example, classical multiscale blob and ridge detectors, such as [10] (Fig. 1(g)), yield many spurious parts, a challenging form of noise for any graph-based indexing or matching strategy. And even if an opportunistic setting of a region segmenter's parameters yields a decent object silhouette (Fig. 1(h)), the resulting skeleton may exhibit spurious branches and may fail to clearly delineate the part structure. From a cluttered image, our two-phase approach recovers, abstracts, and groups a set of medial branches into an approximation to an object's skeletal part structure, enabling the application of skeleton-based categorization systems to more realistic imagery.

2. Related Work

The use of symmetry as a basis for part extraction has a long history in computer vision, including Blum's medial axis transform (MAT) [3], Binford's generalized cylinders [2], Pentland's superquadric ellipsoids [13], and Biederman's geons [1], to name just a few examples. The literature is vast, and space permits us to highlight only a small subset of approaches that assume a 2-D symmetry-based, part-based shape prior *without* assuming an object prior. Thus, approaches that learn to segment particular categories of objects or scenes, often referred to as *image labeling* or *knowledge-based segmentation*, are excluded for they assume knowledge of object or scene content. Likewise, the rich body of skeletonization literature that assumes that a closed curve is provided is also not reviewed here, for it assumes figure-ground segmentation. We thus review only approaches that attempt to extract and group a set of 2-D symmetric parts from a cluttered image.

The use of symmetry as a basis for multiscale abstract part extraction was proposed by Crowley [7], who detected peaks (rotational symmetries) and ridges (elongated symmetries) as local maxima in a Laplacian pyramid, linked together by spatial overlap to form a tree structure. Object matching was then formulated as comparing paths through two trees. Shokoufandeh et al. [18] proposed a more elaborate matching framework based on Lindeberg's multiscale blob model [10]. This family of approaches can be characterized as imposing a strong part-based symmetry prior, detecting parts at multiple scales, and grouping them based on a simple model of spatial proximity. However, simply detecting parts as local maxima in a set of multiscale filter responses leads to many false positives and false negatives, suggesting that successful part extraction requires paying closer attention to image contours.

Symmetry has long been a foundational non-accidental feature in the perceptual grouping community. Many computational models exist for symmetry-based grouping, including Brady and Asada [5], Cham and Cipolla [6], Saint-Marc et al. [15], Ylä-Jääski and Ade [21] and, more re-

cently, Stahl and Wang [20], among others. Such systems face one or more important limitations: 1) the complexity of pairwise contour grouping to detect symmetry-related contour pairs; 2) the requirements of contour smoothness and precise pointwise correspondence dictated by the geometric emphasis of many such approaches; and 3) that such approaches typically stop short of grouping the detected symmetries (parts) into objects.

Our methodology addresses each of these limitations. On the complexity issue, by adopting a region-based approach, our superpixels (medial point hypotheses) effectively group together nearby contours that enclose a region of homogeneous appearance. Drawing on the concept of extracting blobs at multiple scales, symmetric parts will map to "chains" of medial points sampled at their appropriate scale. Our goal will be to group together the members of such chains, ignoring those superpixels (the vast majority) that don't represent good medial point hypotheses. On the smoothness and precision issue, we will learn from noisy training data the probability that two adjacent superpixels represent medial point approximations that belong to the same symmetric part; this probability forms the basis for our affinity function used to cluster medial points into chains. Finally, on the issue of part grouping, we will also learn from noisy training data the affinity function that will form the basis of part attachment. Addressing these three issues yields a novel framework that aims to narrow the gap between work in the segmentation and medial axis extraction communities.

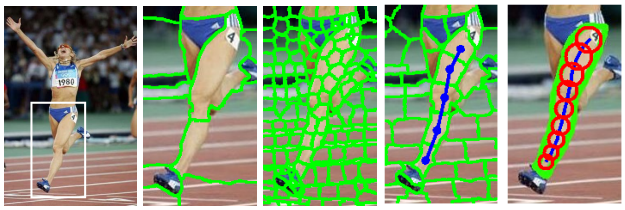
3. Medial Part Detection

The first phase of our algorithm detects medial parts by hypothesizing a sparse set of multiscale medial hypotheses and grouping those that are non-accidentally related. In the following subsections, we detail the two components.

3.1. Hypothesizing Medial Points

Medial point hypotheses are generated by compact superpixels which, on one hand, adapt to boundary structure, while on the other hand, enforce a weak compactness shape constraint. In this way, superpixels whose scale is comparable to the width of a part can be seen as deformable maximal disks, "pushing out" toward part boundaries while maintaining compactness. If the superpixels are sampled too finely or too coarsely for a given part, they will not relate together the opposing boundaries of a symmetric part, and represent poor medial point hypotheses. Thus, we generate compact superpixels at a number of resolutions corresponding to the different scales at which we expect parts to occur; as can be seen in Fig. 1(b), we segment an image into 25, 50, 100 and 200 superpixels. To generate superpixels at each scale, we employ a modified version [12] of the normalized cuts algorithm [17] since it yields compact superpixels.

Each superpixel segmentation yields a superpixel graph, where nodes represent superpixels and edges represent superpixel adjacencies. If a superpixel represents a good medial point hypothesis, it will extend to (and follow) the opposing boundaries of a symmetric part, effectively coupling the two boundaries through two key forms of perceptual grouping: 1) *continuity*, where the intervening region must be locally homogeneous in appearance; and 2) *symmetry*, in that the notion of maximal disk bitangency translates to two opposing sections of a superpixel's boundary. Fig. 2(b) illustrates a symmetry section (blow-up of the subimage in Fig. 2(a) containing the athlete's leg) whose medial point hypotheses are too large (undersampled), while in Fig. 2(c), the medial point hypotheses are too small (oversampled). When they are correctly sampled, as in Fig. 2(d), they can be viewed as a sparse approximation to the locus of medial points making up a skeletal branch, as seen in Fig. 2(e).



a b c d e

Figure 2. Superpixels as medial point samples: (a) a region of interest focusing on the athlete's leg (b) superpixels undersample the scale of the symmetric part; (c) superpixels oversample the scale of the symmetric part; (d) superpixels appropriately sample the scale of the symmetric part, non-accidentally relating, through continuity and symmetry, the two opposing contours of the part; (e) the medial point hypotheses that effectively capture the scale of the part represent a sparse approximation to the locus of medial points that comprise the traditional skeleton.

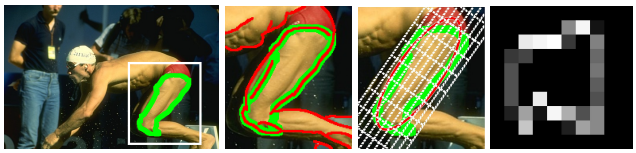
3.2. Clustering Medial Points

If two adjacent superpixels represent two medial points belonging to the same symmetric section, they can be combined to extend the symmetry. This is the basis for defining the edge weights in the superpixel graph corresponding to each resolution. Specifically, the affinity between two adjacent superpixels represents the probability that their corresponding medial point hypotheses not only capture non-accidental relations between the two boundaries, but that they represent medial points that belong to the same skeletal branch. Given these affinities, a standard graph-based clustering algorithm applied independently to each scale yields clusters of medial points, each representing a medial branch at that scale. In Section 4, we group nonaccidentally related medial branches by object, yielding an approximation to an

object's skeletal part structure.

The affinity $A_s(i,j)$ between two adjacent superpixels R_i and R_j at a given scale has both shape A_{shape} and appearance components $A_{appearance}$. We learn the components and how to combine them from training data. To generate training examples, we segment an image into superpixels at multiple scales, and identify adjacent superpixels that represent medial points that belong to the same medial branch as positive evidence; negative pairs are samples in which one or both medial point hypotheses are incorrect or, if both are valid medial points, belong to different but adjacent parts. The boundary of the union of each superpixel pair defines a hypothesized boundary in the image (which may or may not have local gradient support).

To compute the shape-based affinity, we fit an ellipse to the boundary of the union of two adjacent superpixels. We assign an edge strength to each boundary pixel equal to its Pb score [11] in the original image. Each boundary pixel is mapped to a normalized coordinate system defined by the major and minor axes of the fitted ellipse, yielding a scale- and orientation-invariant representation of the region boundary. We compute a 2-D histogram (currently 10×10) on the normalized boundary coordinates weighted by the edge strength of each boundary pixel. This yields a shape context-like feature that reflects the distribution of edges along the presumed boundary of adjacent superpixels. Fig. 3 illustrates the shape feature computed for the superpixel pair from Fig. 1(c), corresponding to the thigh of the swimmer.



a b c d

Figure 3. Superpixel shape feature: (a) boundary of two adjacent superpixels representing two medial point hypotheses; (b) a blow-up of the two superpixels, in which the boundary of their union (green) defines an underlying image edge distribution (red); (c) the normalized scale- and orientation-invariant coordinate system (grid in white) based on the ellipse (red) fitted to the superpixel union; (d) the shape-context-like feature that projects image edgels, weighted by edge strength, into this coordinate system.

We train a classifier on this 100-dimensional feature using our manually labeled superpixel pairs. The margin from the classifier (an SVM with RBF kernel) is fed into a logistic regressor in order to obtain the shape affinity $A_{shape}(R_1, R_2)$ whose range is $[0, 1]$. Table 1 compares various approaches for computing the shape affinity; the SVM with RBF kernel and the SVM with a histogram intersection kernel yield the highest performance.

For the appearance component of the affinity, we compute the absolute difference in mean RGB color, absolute difference in mean HSV color, RGB and HSV color variances of both regions, and histogram distance in HSV space, yielding a 27-dimensional appearance feature. To improve classification, we compute quadratic kernel features, resulting in a 406-dimensional appearance feature. We train a logistic regressor with L1-regularization to yield an appearance affinity measure between two regions ($A_{appearance}(R_1, R_2)$). Training the appearance affinity is easier than training the shape affinity. For positive examples, we choose pairs of adjacent superpixels that are contained inside a figure in the figure-ground segmentation, whereas for negative examples, we choose pairs of adjacent superpixels that span figure-ground boundaries.

	SVM-R	SVM-H	CC	HI
$F_{measure}$	0.75	0.75	0.42	0.44
Mean Precision	0.79	0.79	0.29	0.31

Table 1. Shape affinity comparison according to two measures: $F_{measure}$ and mean precision evaluated on test pairs of superpixels. We evaluate 4 methods: SVM with RBF kernel (SVM-R), SVM with histogram intersection kernel (SVM-H), as well as cross correlation (CC) and histogram intersection (HI) against a mean histogram of all positive training pairs.

We combine the shape and appearance affinities using a logistic regressor to obtain the final pairwise region affinity, converted to edge weights as $W(i,j) = \frac{1}{A_s(i,j)}$. The resulting graph is used in conjunction with an efficient agglomerative clustering algorithm based on [8] (complexity: $O(|S|)$, where S is a set of all superpixels) to obtain medial parts (medial point clusters). The clustering algorithm initializes all medial point hypotheses as singletons, and maintains a global priority queue of edges by decreasing affinity A_s. At each iteration, the highest affinity edge is removed from the queue, and the two clusters that span the edge are hypothesized as belonging to the same part. If each of the two clusters is a singleton, these are merged if the affinity is sufficiently high (the affinity captures the degree to which the union is symmetric). If one or both clusters contain multiple medial points (superpixels), the global symmetry A_s of the union is verified (in the same manner as a pair is verified) before the merge is accepted. Thus, while local affinities define the order in which parts are grown, more global information on part symmetry actually governs their growth. The result is a set of parts from each scale, where each part defines a set of medial points (superpixels). Combining the parts from all scales, we obtain the set $Part_1, Part_2, \ldots, Part_n$. Fig. 1(d) shows the parts extracted at four scales.

4. Assembling the Medial Parts

Medial part detection yields a set of skeletal branches at different scales. The goal of grouping is to assemble the medial branches that belong to the same object. Drawing on the non-accidental relation of proximity, we define a single graph over the union of elements computed at all scales, with nodes representing medial parts and edges linking pairs in close proximity. Assigned to each edge will be an affinity that reflects the likelihood that the two nearby parts are not only part of the same object, but attached. The same graph-based clustering used to detect medial point clusters is used to detect part clusters. However, since some parts may be redundant across scales, a final selection step is applied to yield the final cluster of medial branches, representing an approximation to the object's skeletal part structure. The following two subsections describe the two steps.

4.1. Medial Part Clustering

A minimal requirement for clustering two parts is their close proximity. While the projections of two attached parts in 3-D must be adjacent in 2-D (if both are visible), the converse is not necessarily true, i.e., adjacency in 2-D does not necessarily imply attachment in 3-D (e.g., occlusion). Still, the space of possible part attachments can be first pruned to those that *may* be attached in 3-D. Two parts are hypothesized as attached if one overlaps a scale-invariant dilation of the other (the part is dilated by the size of the minor axis of the ellipse fitted to it, in our implementation).

The edges in the graph can be seen as weak local attachment hypotheses. We seek edge affinities that better reflect the probability of real attachments. We learn the affinity function from training data – in this case, a set of ground truth parts and their attachments, labeled in an image training set. For each training image, we detect parts at multiple scales, hypothesize connections (i.e., form the graph), and map detected parts into the image of ground truth parts, retaining those parts that have good overlap with ground truth. Positive training example pairs consist of two adjacent detected parts (joined by an edge in the graph) that map to attached parts in the ground truth. Negative training example pairs consist of two adjacent detected parts that map to non-attached (while still possibly adjacent in 2-D) parts in the ground truth.

As mentioned earlier, our multiscale part detection algorithm may yield redundant parts, detected at different scales, but covering the same object entity. One solution would be to assign low affinities between such parts. However, in a greedy clustering approach, this would mean that only one part in a redundant set could be added to any given cluster, making the cluster more sensitive to the order in which parts are added. The decision as to which part in a redundant set survives in a cluster is an important one that is best made in the context of the entire cluster. Therefore, we assign a high affinity between redundant parts, and deal with the issue in a separate part selection step.

Formally, our part affinity is defined as:

$$A_p(i,j) = P_r(i,j) + (1 - P_r(i,j))A_{p,\neg r}(i,j) \quad (1)$$

where $P_r(i,j)$ is the probability that parts i and j are redundant, and $A_{p,\neg r}(i,j)$ is the affinity between the parts given non-redundancy. $P_r(i,j)$ is computed by training a quadratic logistic classifier over a number of features, including overlap (in area) of the two parts (O_{ij}), defined as the overlap area normalized by the area of the smaller part, overlap of the two parts' boundaries (B_{ij}), and appearance similarity (A_{ij}) of the two parts. The features are defined as follows:

$$\begin{aligned} O_{ij} &= \frac{|Part_i \cap Part_j|}{\min\{|Part_i|, |Part_j|\}} \\ B_{ij} &= \frac{|\partial(Part_i \cap Part_j)|}{\min\{|\partial Part_i|, |\partial Part_j|\}} \\ A_{ij} &= A_{appearance}(Part_i, Part_j) \end{aligned} \quad (2)$$

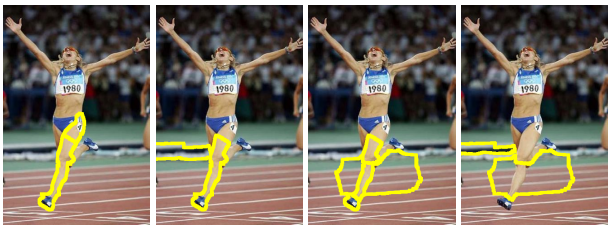
where $|\cdot|$ is the region area and $|\partial(\cdot)|$ is the region perimeter.

The affinity $A_{p,\neg r}(i,j)$ between non-redundant parts i and j, like affinities between medial points, includes both shape and appearance components. The components are best analyzed based on how the two parts are attached. Given an elliptical approximation to each part, we first compute the intersection of their major axes. The location is normalized by the length of the major axis, to yield a scale-invariant attachment position r for each part. We define three qualitative attachment "regions" to distinguish between four attachment types: inside ($|r| < 0.5$) , endpoint ($0.5 < |r| < 1.5$), or outside ($|r| > 1.5$). Our four apparent attachment categories can be specified as follows:

1. end-to-end ($J_{ij} = 1$) – The intersection lies in the endpoint region of both parts.
2. end-to-side ($J_{ij} = 2$) – The intersection lies in the inside region of one part and in the endpoint region of the other part.
3. crossing ($J_{ij} = 3$) – The intersection lies in the inside region of both parts.
4. non-attached ($J_{ij} = 4$) – The intersection lies in the outside region of one or both parts.

Fig. 4 gives examples of these four attachment types.

The shape component of our affinity is based on the principle that when one part attaches to (interpenetrates) another, it introduces a pair of concave discontinuities (Hoffman and Richards' principle of transversality [9]), reflected as a pair of L-junctions marking the attachment. In contrast, when one part occludes another, the L-junctions are



$J_{ij} = 1$ $J_{ij} = 2$ $J_{ij} = 3$ $J_{ij} = 4$

Figure 4. Attachment categories. The four different attachment categories of parts (yellow).

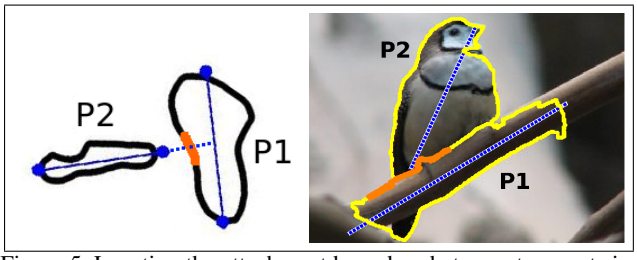


Figure 5. Locating the attachment boundary between two parts in the case of an end-to-side attachment. The attachment boundary (orange) between the two parts P_1 and P_2 is centered at the intersection of the major axis of P_2 with the boundary of P_1, and extends along the boundary of P_1 a total distance equal to the length of the minor axis of P_2. Left - illustration of the attachment boundary. Right - attachment boundary between two parts in a real image.

replaced by T-junctions, reflecting an occlusion boundary. This is a heuristic, for there could be an appearance boundary between two attached parts, misinterpreted as an occlusion boundary.

Since extracting and classifying contour junctions is challenging in the presence of noise, we will instead focus on the evidence of an occlusion boundary between two parts, based on image edges (E_{ij}) along the attachment boundary between parts i and j. Once the attachment boundary is found, evidence is accumulated as the average Pb [11] of the boundary pixels. Finding the attachment boundary is not trivial since the parts may be sufficiently close but not touching, due to segmentation errors.

The attachment boundary is computed similarly for all four attachment categories. For a pair of attached parts, we first select the part P_1 with the smaller $|r|$ and find the intersection of its boundary with the major axis of the other part P_2. The attachment boundary is centered at the intersection and extends along the boundary of P_1 in both directions, to an extent equal to the length of the minor axis (width) of P_2. For end-to-side attachments, this is illustrated in Fig. 5.

Given the attachment category J_{ij}, the attachment boundary evidence E_{ij}, and the appearance similarity A_{ij}, we can define the part affinity $A_{p,\neg r}(i,j)$. One logistic classifier is trained for end-to-end junctions ($A_1(i,j)$), whereas another is trained for end-to-side junctions ($A_2(i,j)$). For crossing and non-attached junctions, we set the affinity to 0 because we empirically found that none of the high-affinity part pairs in the training set exhibited such attachment categories. Our affinity for non-redundant parts becomes:

$$A_{p,\neg r}(i,j) = [J_{ij} = 1] \cdot A_1(i,j) + [J_{ij} = 2] \cdot A_2(i,j) \quad (3)$$

Having defined all the components of the affinity function $A_p(i,j)$ (Equation 1), we use these affinities to cluster parts that are attached. We use the same algorithm [8] used to cluster medial points into parts.

4.2. Medial Part Selection

Our affinity-based grouping yields a set of part clusters, each presumed to correspond to a set of attached parts belonging to a single object. However, any given cluster may contain one or more redundant parts. While such parts clearly belong to the same object, we prune redundancies to produce the final approximation to an object's skeletal part structure. Our objective function selects a minimal number of parts from each cluster that cover the largest amount of image, while at the same time minimizing overlap between the parts. The problem is formulated as minimizing a quadratic energy over binary variables. Let $X_i \in \{0,1\}$ be an indicator variable representing the presence of the i^{th} part in a cluster. We seek the subset of parts that minimizes the following energy:

$$E = \sum_i X_i \left(K - |Part_i|\right) + \sum_{i,j} X_i X_j O_{ij} \quad (4)$$

where K controls the penalty of adding parts. In our experiments, we found that $K = 0.1 \cdot median\{|Part_i|\}$ is an effective setting for this parameter. We find the optimal X by solving a relaxed quadratic programming problem, in which real values are rounded to 0 or 1 [14].

5. Results

To evaluate the method, we train the various components using the Weizmann Horse Database [4], consisting of images of horses together with figure-ground segmentations; in addition, we manually mark the elongated parts of the horses, together with their attachment relations. Fig. 6 illustrates an example training image and its ground truth segmentations. Once trained, we first qualitatively evaluate the system on images of objects with well-defined symmetric parts drawn from *different* (i.e., non-horse) image domains, reflecting our assumption that both symmetry and part attachment are highly generic.

Fig. 7 shows the results of our algorithm applied to a number of different image domains. In each case (a-h), the figure shows one or two of the most prominent groupings of medial branches. The abstractions of the parts in each

Figure 6. Ground truth used for training: sample image (left), figure/ground segmentation (middle), and part segmentation (right).

cluster are shown as ellipses with their major axes (medial branch regularizations) depicted by dotted lines.[1] All other parts are shown with faint (grey) elliptical part abstractions (without axes, for clarity), illustrating the ability of our algorithm to correctly group medial branches.

Examining the results, we see that in Fig. 7(a), our system has successfully extracted the major parts of the athlete and correctly grouped them together. Fig. 7(b) illustrates not only that the parts of the windmill were successfully recovered and clustered, but that the person was also recovered as a separate single-part cluster. The smaller windmills undetected in the background contain parts whose scale was smaller than our finest sampled scale. Figs. 7(e,f,g) show other examples of our system's success, in which the major medial parts of a plane, swan, and statue, respectively, were recovered and grouped to yield an approximation to an object's skeletal part structure.

Figs. 7(c,d, and h) illustrate some limitations of our approach. For example, in Fig. 7(c), one of the swan's wings is not properly detected. Due to insufficient contrast between the wing and the background, the superpixel boundaries fail to capture the part at any scale. Still, the remaining part structure of the swan may provide a sufficiently powerful shape index to select a small set of candidate models, including the swan model, which could be used in a top-down manner to overcome such segmentation problems. In Fig. 7(d), a more serious problem occurs when too many parts are clustered due to a lack of contrast at their attachment boundaries (symmetric strip of horizon landscape accidentally grouped with vertical mast). Like Fig. 7(c), a candidate model may be required to resolve such ambiguous part attachments. Finally, Fig. 7(h) shows that although the main parts of the lizard are found, the tail is not composed of a single part since our system assumes parts with straight symmetry axes.

Finally, to provide a quantitative evaluation of our part detection strategy, we compare its precision and recall to the method of Lindeberg et al. [10], used to generate the symmetric parts shown in Fig. 1(g). Both methods are evaluated on 61 test images from the Weizmann Horse Dataset [4]. A ground truth part is considered to be recovered if its normalized overlap (in area) with one of the detected parts is above a threshold (0.4). Our part detection offers a significant improvement in both precision and recall (Fig. 8). Moreover, in [10], no effort is made to distinguish part occlusion from part attachment; parts are simply grouped if they overlap. Note that both methods achieve low precision. This is partially due to the fact that there are other symmetric parts in the images, besides the horses' parts, that were not marked in the ground truth.

[1] Note that although we choose to display only the abstract representation of a part, the medial points that make up the part are available. In turn, each medial point defines a location (centroid) and radius (of best fitting maximal disk centered at the centroid), providing a more standard skeletal description.

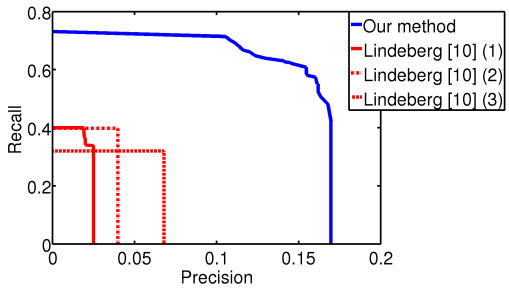


Figure 8. Precision vs recall of part detection. Due to the low precision of [10], we prune small parts to increase precision: (1) no pruning, (2) prune parts whose major axis is less than 10 pixels, (3) prune parts whose major axis is less than 20 pixels.

Limitations and future work: A number of limitations of the current framework will be addressed in future research. To improve the quality of medial point hypotheses, we are exploring a more powerful superpixel extraction framework that allows greater control over compactness, along with a multiscale Pb detector. We also intend to relax our linear axis model to include curved shapes; for example, the ellipse model could easily be replaced by a deformable superellipse. Finally, at the part grouping level, we aim to address the problem of object undersegmentation (false positives) through the incorporation of closure constraints.

6. Conclusions

We have presented a constructive approach to detecting symmetric parts at different scales by grouping small compact regions that can be interpreted as deformable versions of maximal disks whose centers make up a skeletal branch. In this way, symmetry is *integrated* into the region segmentation process through a compactness constraint, while region *merging* is driven by a symmetry-based affinity learned from training data. Detected parts are assembled into objects by exploiting the regularities of part attachments in supervised training data. The resulting framework can recover a skeletal-like decomposition of an object from real images without requiring any prior knowledge of scene content and without requiring figure-ground segmentation.

Acknowledgements: This work has been supported, in part, by the European Commission, under a Marie Curie Excellence Grant (MCEXT-025481) to C.S.

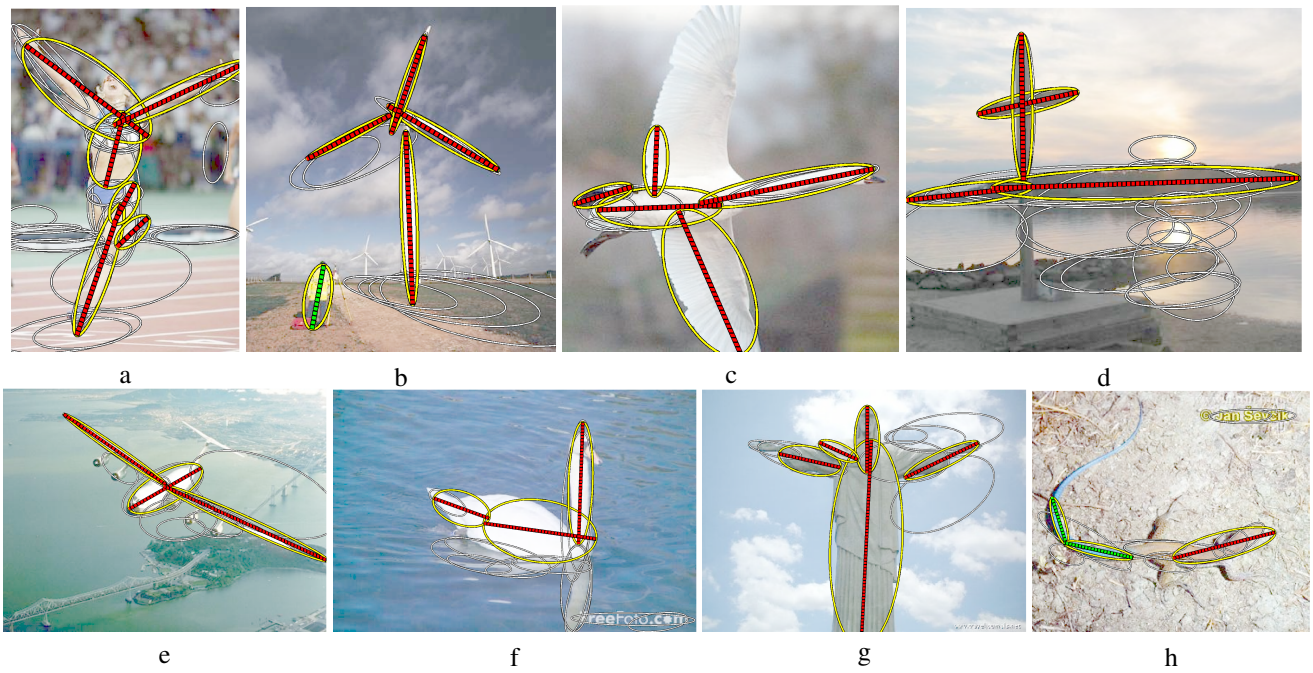


Figure 7. Detected medial parts and their their clusters. In each image, we show the most prominent cluster, showing the medial branch (red dashed) and extent (yellow ellipse) of each abstract part. In some images, a secondary part cluster is shown with green medial branches. All other parts are shown faintly in grey.

References

[1] I. Biederman. Human image understanding: Recent research and a theory. *CVGIP*, 32:29–73, 1985. 2

[2] T. O. Binford. Visual perception by computer. In *Proceedings, IEEE Conference on Systems and Control*, Miami, FL, 1971. 2

[3] H. Blum. A Transformation for Extracting New Descriptors of Shape. In W. Wathen-Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380. MIT Press, Cambridge, 1967. 1, 2

[4] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV*, pages 109–124, 2002. 6, 7

[5] M. Brady and H. Asada. Smoothed local symmetries and their implementation. *IJRR*, 3(3):36–61, 1984. 2

[6] T.-J. Cham and R. Cipolla. Geometric saliency of curve correspondences and grouping of symmetric contours. In *ECCV*, pages 385–398, 1996. 2

[7] J. Crowley and A. Parker. A representation for shape based on peaks and ridges in the difference of low-pass transform. *PAMI*, 6(2):156–169, March 1984. 2

[8] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004. 4, 6

[9] D. D. Hoffman, W. Richards, A. Pentland, J. Rubin, and J. Scheuhammer. Parts of recognition. *Cognition*, 18:65–96, 1984. 5

[10] T. Lindeberg and L. Bretzner. Real-time scale selection in hybrid multi-scale representations. In *Scale-Space*, volume 2695 of *Springer LNCS*, pages 148–163, 2003. 2, 7

[11] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26:530–549, 2004. 4, 6

[12] G. Mori, X. Ren, A. A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *CVPR*, pages 326–333, 2004. 3

[13] A. Pentland. Perceptual organization and the representation of natural form. *AI*, 28:293–331, 1986. 2

[14] A. P. Pentland. Automatic extraction of deformable part models. *IJCV*, 4(2):107–126, 1990. 6

[15] P. Saint-Marc, H. Rom, and G. Medioni. B-spline contour representation and symmetry detection. *PAMI*, 15(11):1191–1197, 1993. 2

[16] T. Sebastian, P. N. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *PAMI*, 26(5):550–571, 2004. 1

[17] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000. 2, 3

[18] A. Shokoufandeh, L. Bretzner, D. Macrini, M. F. Demirci, C. Jönsson, and S. Dickinson. The representation and matching of categorical shape. *CVIU*, 103(2):139–154, 2006. 2

[19] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker. Shock graphs and shape matching. *IJCV*, 30:1–24, 1999. 1

[20] J. Stahl and S. Wang. Globally optimal grouping for symmetric closed boundaries by combining boundary and region information. *PAMI*, 30(3):395–411, 2008. 3

[21] A. Ylä-Jääski and F. Ade. Grouping symmetrical structures for object segmentation and description. *CVIU*, 63(3):399–417, 1996. 2