Category-Specific Object Recognition and Segmentation Using a Skeletal Shape Model

Nhon H. Trinh
ntrinh@lems.brown.edu
Benjamin B. Kimia
kimia@lems.brown.edu
Brown University
Providence, RI 02912 USA

Abstract

The success of skeletal model in object recognition from segmented images motivates the development of a skeletal model for top-down object recognition and segmentation. We propose a novel skeleton-based generative shape model which is suitable for efficient search using dynamic programming (DP). We have devised an exclusion principle enabling DP to discover multiple instances of an object category in one pass. Finally, we have improved an oriented chamfer distance for rank-ordering generated hypotheses. Improved or comparable recognition and segmentation results are reported on the ETHZ data set.

1 Introduction

Category-specific recognition and segmentation generalize appearance or shape cues. Appearance based methods generally rely on feature points [34, 35] and have had remarkable success in detecting the presence of objects [3, 7, 10, 14, 24, 28, 30, 31, 47, 52, 56] and in object localization [3, 28, 31, 53, 55]. However, these methods can be limited because stable key features are not always abundantly available, *e.g.*, in low lighting condition, in bright backgrounds, objects with large homogeneous patches, objects in low resolution imagery [40]. More importantly the role of appearance may become severely diminished as the intra-category variation increases with respect to inter-category variation. For example, in recognizing bottles and cups the surface markings are simply too varied to be useful [37]. In these cases, the silhouette and the internal markings which are consistent across the category become the primary source of information for recognition. In general, it can be argued that **function** is more dependent on object **form** than its **appearance** [32, 48, 49, 50], thus motivating augmenting the role of appearance with shape.

Two interconnected key issues emerge in the use of form for recognition, which motivated a skeletal representation. First, observe a **repeatability/distinctiveness** tradeoff in the choice of feature: the more complex a contour feature is the more chance it has to merge with background clutter, but at the same time it is more distinctive in the selection of a category and can also delineate the object better. For example, **simple edges** have been used in annular regions [23, 25], in pairwise relations [32], and in a deformable, non-rigid point matching approach [2, 3, 5]. Two independently developed models use code books of **boundary fragments** learnt from training data under boosting [36, 37, 38, 45, 46]. The

BMVC 2009 doi:10.5244/C.23.6

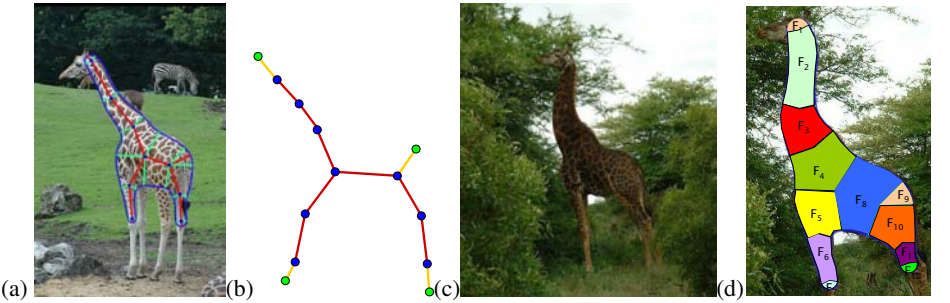


Figure 1: Exemplar shapes (a) define prototypical shock graph topologies (b) and a range of metric variation which when searched over a new image (c) can detect and delineate category instances (d).

k-adjacent segments (kAS) model uses paths of length k on a contour segment network of grouped and polygonized Berkeley edge map as basic features [15, 17], with a pair $k = 2$ being optimal (PAS). This paper suggests the use of an even more complex feature, a **visual fragment** [26], which is a pair of contour fragments corresponding to a skeletal segment together with the region between them, Figure 2. This use of a pair of contours is more general than PAS [17], as they are not restricted to be adjacent, and can represent non-linear curve geometry. They also retain order along contour in contrast to [45] which is an unorganized collection of edges. In addition to contour geometry, visual fragments also represent the region that is sandwiched between two contour fragments, thus increasing the distinctiveness of this feature for category-level recognition.

A second issue is on the use of a shape model to represent the **spatial relationship** among features whose role becomes more critical the simpler the feature is. The use of extrinsic deformations such as thin plate spline (TPS) and shape context in matching [2, 5, 16, 18, 33, 44, 57] is a popular method of representing relative spatial relationship. However shape context does not behave well under large deformation such as articulation and occlusion. The star model of [36, 37, 45, 46] is restricted to rigid objects and also suffer under articulation since training with articulated objects would smear the proposed object center, *e.g.*, the head of a horse or giraffe in relation to its body. Finally, training for boundary fragment codebook can only handle a modest degree of shape deformation, as is the case with Geometric Hashing [29] and Hough transform. In contrast, the skeletal/shock graph model is robust under articulation and bending as evidenced by the precision-recall curve comparing it to the shape context [43]. The shock graph representation of a closed shape is a distributed pictorial structure-type representation so that it does not rely on single object center, and it can handle a large variety of shape deformations.

Our approach to category-specific object recognition and segmentation, Figure 1, relies on a number of components. First, it requires the availability of a **generative shape model** *(i)* capable of accurately describing a wide range of free-form shapes, *(ii)* featuring a degree of invariance with within-category variations, and *(iii)* achieving this via a low-dimensional space. Second, it requires an **efficient search** methodology so that generated shape hypotheses can be examined against the image and rank-ordered. Third, the ranking-ordering requires an **objective function** to capture the extent of support each hypothesis receives from the image. We given an overview of each in turn.

First, a generative model for shape based on the shock graph is proposed in [54], where given a fixed shock graph topology, shapes with piecewise circular arc boundaries are generated from a low-dimensional parametric space, Figure 2. While this generative model is

highly suitable for object recognition, the intrinsic nature of the representation from an extrinsic reference node leads fragments to be dependent on the chain fragments back to the reference node. This inter-dependence of fragments forces combinatorial search in the segmentation application. A contribution of this work is to define a generative model of shape where fragments are **not** dependent on each other, Section 2. Given a number of exemplars, category prototypes are generated by capturing the common shock graph topologies and a metric range associated with each. This allows for the generation of synthetic examples, Section 3.

Second, the decomposition of shape into subshapes at nodes allows for a dynamic programming (DP) approach to search for the optimal shape hypothesis in the image. However the traditional DP cannot find multiple category instances in the same image since it only gives the globally optimal solution. We devise an exclusion principle that allows for detecting multiple category instances without having to run DP multiple times. Third, the traditional oriented chamfer matching is modified to improve on over-counting and under-counting drawbacks, as well as to include a more global penalty for groups of edges. In addition, the cost function is tailored to each specific fragment based on observation in the exemplar space.

Finally when the three component come together, the presence of object category instances can be identified and delineated in the image. Object recognition performance on the ETHZ dataset shows improvement on three categories, and is comparable for the remaining two, the mug and apple logo. We have also assessed segmentation coverage/precision, showing improvements in the same three categories, Section 6.

2 Fragment-Based Generative Model for Shape

The large within-category shape variation can be effectively captured by a variant of the medial axis, the shock graph: shock graph topology describes the shape *qualitatively* while shock segment metric attributes describe the shape boundary in details. Trinh and Kimia [54] proposed a generative model to synthesize shapes sharing a given shock graph topology from a fairly low number of parameters. The model approximates a shape boundary with a piecewise circular arc spline, which is dense in the space of smooth contours, and breaks a shape into **fragments**, each of which is bounded by a pair of circular arcs, Figure 2a. Utilizing the continuity constraints between adjacent fragments to reduce away dependent parameters, the model devises a set of *independent parameters* capturing all variations of shapes sharing the given shock graph topology. The parameter set consists of: *(i)* the extrinsic parameters residing at one "reference" node encoding the position, orientation and size of the object and *(ii)* the intrinsic parameters distributed among the nodes and edges of the graph, encoding the local intrinsic properties of the shape, *e.g.*, stretching, bending, bulging.

The separation into intrinsic and extrinsic parameters makes the model powerful for object recognition since bending or stretching at any part of the shape corresponds to changing just one or two intrinsic parameters. However, this arrangement also results in an *inter-dependence* among fragments since describing each fragment's boundaries requires all the extrinsic and intrinsic parameters along the fragment chain back to the "reference node." This dependency renders the model unsuitable for object segmentation as it leads to combinatorial search.

We propose a model where each fragment is fully described locally by the parameters at its adjacent nodes, eliminating the inter-dependency between the fragments. Specifically, a

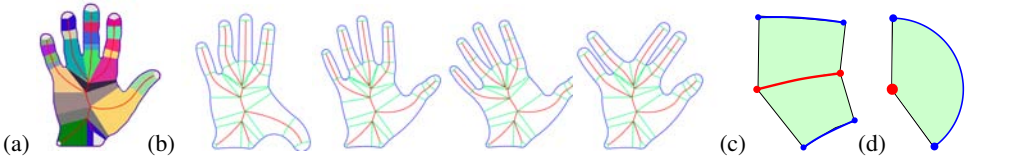


Figure 2: Trinh and Kimia's shape model [54]: (a) A shape boundary is approximated with a closed piecewise circular arc spline (PCAS) and is decomposed into "shape fragments" distributed along its medial axis. (b) Shape articulation and other variations can be realized by changing just a few parameters. (c) An A_1^2 fragment. (d) An A_∞ fragment.

shape is represented by its shock graph topology, $G = (V,E)$ where $V = \{v_0, v_1, v_2, \cdots, v_N\}$ and $E = \{e_1, e_2, \cdots, e_M\}$ are the sets of shock nodes and shock edges, respectively, and where the degree of nodes is generically 1, 2, or 3 [20]. Here we restrict a shape to be a simple closed contour so that the shock graph G is always a (rooted) tree, $M = N$. The shock graph topology naturally decomposes a shape into **fragments**, defined as the influence zone of a shock segment (shock graph edge), *i.e.*, the region bounded by the boundary portions corresponding to the shock segment and the orthogonal rays from the shock nodes to the boundary contours, Figure 2(c-d) . There are two types of fragments in the domain of shapes with circular arc spline boundaries: an A_1^2 fragment corresponding to a curve segment of the shock graph, Figure 2c, and an A_∞ fragment signifying the end of a shock branch, Figure 2d.

Parameter set Fragments can be reconstructed from the local information at their end nodes: an A_1^2 fragment is bounded between two degree-2 nodes and an A_∞ fragment can be considered as a special case of an A_1^2 fragment where the two shock nodes coincide and the two boundary points corresponding to one shock node also coincide. The local first-order geometry of the shape boundary reconstructed from any shock node v_i requires the following [20]

$$Z = \{z_i = (x_i, y_i, \psi_i, r_i, \Phi_i), \quad i = 1, \cdots, N\}, \tag{1}$$

where (x_i, y_i) specifies the position of the node v_i, ψ_i is the angle of the shock tangent at v_i, pointing away from its parent edge, r_i is the radius at the shock node, and Φ_i is the φ-angles of shock segments coincident at v_i, Figure 3a-b. The boundary contour of each fragment can be reconstructed from the parameters of the two nodes adjacent to it, Figure 3c. The 5 parameters at a degree-2 node determine the positions and orientations of its two boundary points. Similarly, the 6 parameters at a degree-3 node uniquely determine the positions and orientations of its 3 boundary points. This means that the parameter set Z fully specify the positions and orientations of the starting and ending points of both boundary contours of every shape fragment. To reconstruct the fragment's boundary in fullness we interpolate between the starting and ending point using the smooth bi-arc interpolation [42], leading to

Proposition 1. *Given its shock graph topology G, the boundary of a shape with circular arc boundary is fully specified from the parameter set Z as described in (1).*

3 From Exemplars to Prototypes

Since the specification of a shock graph topology can generate shapes well beyond one category, Figure 4a, and since each object category typically maps to a few shock graph topologies, Figure 4b, we use a few prototypes $S = \{s_k = (G_k, g_k),\ k = 1, \cdots, l\}$ to describe shapes

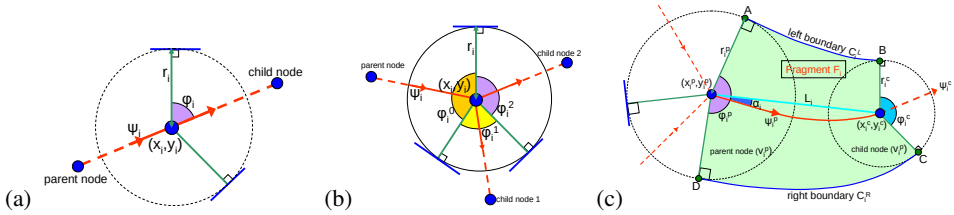


Figure 3: There are 5 parameters stored at each degree-2 node (a) and 6 parameters at each degree-3 node (b). These attributes specified the positions orientations of all boundary points at each node. (c) Smooth interpolation between the point-tangent pairs at each fragment using bi-arcs gives us the boundary contour of the entire shape.

in an object category, where G_k is a shock graph topology, and g_k is a shape prior that captures the allowed variation (likelihood) of the geometric attributes of G_k. In this paper we use a very simple prior: g_k is a uniform distribution over the intrinsic attributes of the shape fragments.

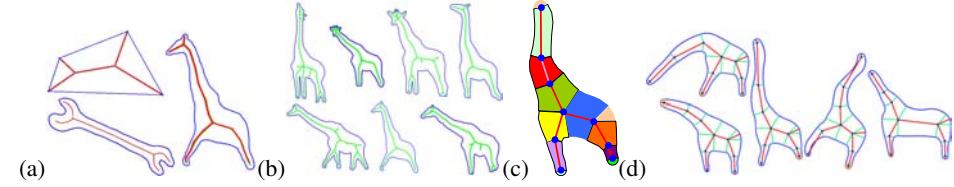


Figure 4: (a) Shapes of different categories may share the same shock graph topology. (b) Category instances can lead to a few distinct shock graph topologies, (c) A giraffe prototype and (d) synthetically generated exemplars, which look like a giraffe.

Training a model on exemplars Given a set of (positive) training images for a specific object category we manually delineate the boundary of exemplars and compute their shock graphs topologies G_i [51] and extract their metric attributes Z_i. We identify shock graph topologies closely shared by many exemplars and manually construct a prototype for each group, Figure 4b. We then generate shapes best fitted to each exemplar by manual selection of the parameter sets, resulting in a set of exemplar parameters for each prototype. The ranges of variation of these exemplars parameters, padded by 50%, are fixed as the allowed range for that prototype, Figure 4c. We can now generate synthetic boundary contours that "look like" shapes of the category of interest, including those not present in the training images, Figure 4d, which despite the rudimentary nature of the distribution and the training procedure, are visually quite satisfactory. We recognize, however, that this very simple model can be strengthened, expecting significant improvements.

4 Single-Pass Multiple Solution Dynamic Programming

We now address the issue of finding shapes with adequate image support among those generated by our model. We select the dynamic programming approach [4, 19] because it gives a global optimal in polynomial time while constraining the cost function in a reasonable way. DP has been used to optimize objective functions defined on a chain [6], a tree [12], and more complex graphs [1, 11, 19].

In our search problem, a shape can be decomposed into at most three pieces (sub-shapes) at a shock node, each corresponding to a subtree rooted at the node. This decomposition of

(a)   (b)  

Figure 5: Images with multiple instances of the same object category when spatially distant (a) and when overlapping (b). Sliding window method faces difficulty in the latter case.

shape into sub-shapes allow DP to be applied as long as the objective function of a shape can be written as a sum of costs of its fragments, a fairly easy requirement to meet, Section 5. Specifically, selecting an arbitrary node, say v_0, as root, each fragment F_i can be enumerated by the index of the child node v_i which has a unique parent node $v_{\hat{i}}$. Thus, denoting the cost of a fragment F_i as $f_i(z_{\hat{i}}, z_i)$, where z_i and $z_{\hat{i}}$ are the states of v_i and $v_{\hat{i}}$, respectively, we have the cost of a shape hypothesis with state Z as

$$f(Z) = \sum_{i=1}^{N} f_i(z_{\hat{i}}, z_i). \tag{2}$$

Denote the optimal cost for all possible child sub-shapes at a node v_i with state z_i as $f_i^*(z_i)$. Let C_i denote the set of all immediate child nodes of z_i. Then equation (2) leads to a recursive formulation

$$f_i^*(z_i) = \sum_{v_j \in C_i} \min_{z_j} \left(f_j(z_i, z_j) + f_j^*(z_j) \right), \tag{3}$$

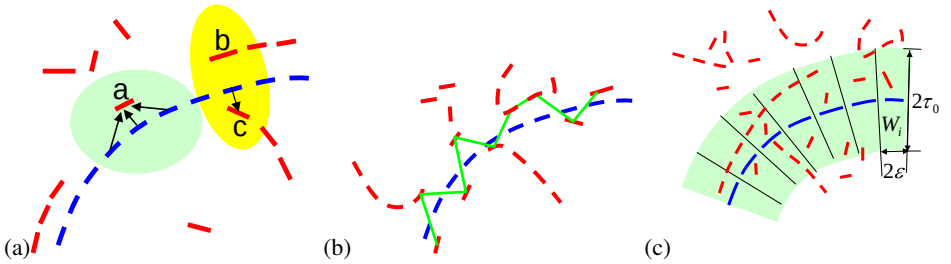
where z_j is the state of v_j. When the node v_i is a leaf node and has no children, we define $f_i^*(z_i) = 0$ and the cost at the root node v_0, $f_0^*(z_0)$, defines the cost of the shape. The recursive Formula (3) allows for an application of DP when the optimal cost of child subshapes are computed in the order of children to parents. The optimal shape can be retrieved by backtracking if at each node the state which achieved optimal child subshape cost is stored.

Exclusion Principles and Multiple-Solution DP: The above algorithm only returns the globally optimal solution and thus fails to return additional instances of a category when present in an image, Figure 5a. A common solution is the use of a *sliding window* where objects are detected in multiple overlapping rectangular boxes with sufficiently dense sampling of sizes and positions [8, 17, 55]. This method finds spatially non-overlapping multiple instances, but will miss those instances overlapping within the same window, Figure 5b.

Our approach first enlarges the space of solutions to one optimal DP solution per root node state and then uses a new criterion to select a limited subset. Specifically, observe that distinct object instances almost always have distinct root node states: for two root node states to be identical requires location, orientation, local scale, and local angles all to match, extremely unlikely for two distinct object instances. The set of all possible root node states thus create an initial solution space of candidate objects. This set is rather large, *e.g.*, 2 million large in a typical case for our setting, and needs to be reduced in size.

Differential Exclusion Principle. *An infinitesimal perturbation of the apparent contour of an object instance in an image does not arise from the projection of another object instance.*

This principle states that it is unlikely that an instance's perturbation to exactly "eclipse" another instance. Together with topological structure of the root node's state space, it allows the global solution at a node to inhibit global solutions at a neighborhood when the objective function judges these lateral solutions to be inferiors, thus requiring



(a) (b) (c)

Figure 6: (a) OCM can overcount edge labeled "a", or may undercount by not using the best edge, edge "c" instead of "b"; (b) Accidental alignment between a contour and the image edges often forms a zig-zagging image contour. (c) In CCM, the best matching edge is restricted to a window around each contour point.

Proposition 2 (Local Minimum Requirement)**.** *The shape hypothesis corresponding to the projection of an object instance is a local minimum of the objective function in the state space of the root node.*

5 Rank-Ordering Hypotheses

We now define the objective function, which is constrained to be sum over image support for individual fragments, but is otherwise generic. Image support for a shape fragment can be based on the region appearance [8] or based on the pair of bounding contours [13, 46]. This paper focuses on the latter in the form of edge-based support but the use of appearance and small contour fragments would likely improve the results.

Previous approaches to capturing edge support for a model have used the Hausdorff metric [21], and *Oriented Chamfer Matching (OCM)* [22, 46]. The OCM cost is a sum over edges of the model contour, each of which is a linear sum of normalized distance (Chamfer distance) and normalized orientation difference with a best matching edge.

The current form of the OCM metric has several drawbacks. First, an image edge can contribute support to multiple contour points, Figure 6a, thus *over-counting* the support. Second, the edges matched to the contour points may not be the best edge available, Figure 6a, especially in scenes with many spurious edges, thus *under-counting* the image support for the query contour. Third, this cost function rewards accidental alignment between edges and the contour, *i.e.*, when the image edges individually are in close proximity and align well with the query contour but they collectively form a contour not resembling the query contour, Figure 6b. These drawbacks reduces the sensitivity of the OCM, especially when the scene contains many spurious edges or missing edges.

Our improvements to OCM: Contour Chamfer Matching(CCM) First, we force each image edge to support at most one contour point by restricting the selection of matching edges to a thin window orthogonal to the contour point, Figure 6c. Second, we match each contour point to its best supporting edge rather than the closest one by searching this thin window for the edge that minimizes the contour point's OCM cost. In addition accidental alignment between the model contour γ and the connected best edge correspondences, $\bar{\gamma}$, is penalized, which "zig-zags" around the model curve.

Formally, let the query contour γ be represented as a set of ordered oriented points $\gamma = \{\gamma_i\}_{i=1}^{N}$. Let $\{e_j\}_{j=1}^{N}$ be the unordered set of edges (the edge map). We select a best matching

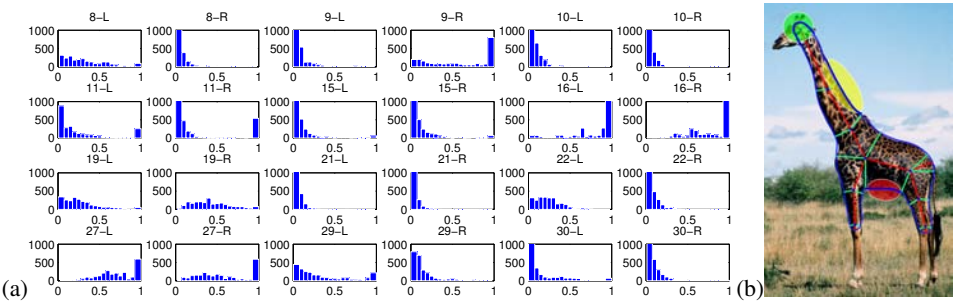


Figure 7: (a) Histogram of CCM costs of contour fragments of the ETHZ giraffe shapes are different because some curve segments are more detectable than others (comparing the neck region (yellow) with the belly region (red) and because certain portions of the boundary, *e.g.*, the head (green), are approximated and thus receive no support.

edge $\bar{\gamma}_i$ for each oriented point γ_i by searching among edges in a window W_i defined by the rectangle with its short side along the edge γ_i extending 2ε and the long side extending τ_0 in each direction, Figure 6c. The cost for matching edges is the oriented chamfer cost,

$$\bar{\gamma}_i = \arg\min_{e_j \in W_i} \left[(1-\lambda_1) \min\left(\frac{d(\gamma_i, e_j)}{\tau_1}, 1 \right) + \lambda_1 \min\left(\frac{|\alpha_i - \bar{\alpha}_j|}{\tau_2}, 1 \right) \right], \tag{4}$$

where α_i is the tangent angle of γ_i and $\bar{\alpha}_j$ is the orientation of the edge e_j, τ_1 is the distance normalization constant, τ_2 is the angle normalization constant, and λ_1 modulates the effect of orientation to the effect of distance. Let $\bar{\beta}_i$ be the tangent angle of the curve $\bar{\gamma}$ at point $\bar{\gamma}_i$, which is typically different from the edge orientation of $\bar{\gamma}_i$, $\bar{\alpha}_i$. Our final cost is defined as

$$d_{\text{CCM}} = \frac{1}{N} \sum_{i=1}^{N} \left[(1-\lambda_1) \min\left(\frac{d(\gamma_i, \bar{\gamma}_i)}{\tau_1}, 1 \right) + \lambda_1 \min\left(\frac{|\alpha_i - \bar{\alpha}_i|}{\tau_2}, 1 \right) + \lambda_2 \min\left(\frac{|\alpha_i - \bar{\beta}_i|}{\tau_2}, 1 \right) \right]. \tag{5}$$

Tailoring the cost function to each specific fragment Observe in Figure 7a that the histograms of the CCM costs on contour segments of a giraffe shape, collected from their corresponding portions ranging over the giraffe images of the ETHZ dataset [15], differ substantially. We hypothesize that the observed differences are due to two factors: First, some curve segments may be more detectable than others due to context, *e.g.*, the contour along the neck of a giraffe is more detectable than that along its belly, Figure 7b. Second, our shape model approximation can at time deviate from the true boundary so it receives no support, green blob in Figure 7b. This motivates transforming the observed cost through the distribution P_γ computed from the training data $w(\gamma) = -\log P_\gamma(d_{CCM}(\gamma))$, which are then summed over the two boundaries of each fragment to reflect the final cost.

6 Experiments

The 255-image ETHZ Shape Classes dataset [15] is used to measure the performance of our algorithm. This is a challenging data set in which objects from 5 categories (apple logo, bottle, giraffe, mug, and swan) vary in size and pose and often blend with cluttered backgrounds. The most challenging is the giraffe category [57] since giraffes articulate and the background is cluttered with texture-rich objects such as trees, leaves, and grass.

In our implementation, edges are first computed using the Pb edge detector [9] and then pruned using Kovesi's edge linker [27] to reduce cluttered edges (removing edges that do not belong to any linked contours of at least 10 pixel long). For the CCM stage, the boundary contour is sampled uniformly with sampling rate $ds = 2$ pixels and parameters are set as $\lambda_1 = 0.3$, $\lambda_2 = 0.4$, $\tau_1 = 8$, $\tau_2 = \frac{\pi}{4}$, and $\tau_0 = 3$. The DP stage uses $\Delta x = 8$, $\Delta y = 8$, $\Delta\phi = \pi/17$, $\Delta\psi = \pi/8$, $\Delta\phi = \pi/16$, $\Delta\sigma = \pi/16$, $\Delta\delta = \pi/16$, $\Delta(\log_2 r) = 0.2$.

Figure 8a shows the object recognition performance of our method (in red) for each of the five categories in comparison to the Contour Selection method [57] in blue, the Automatic Learnt Shape Model method [18] in green (for fairness we compare against their best), and the Multi-stage Contour Based Detection method [41] in cyan. For the methods for which a full precision-recall curve is available, we show significant improvements at least in three categories (bottles, giraffes, and swans), same performance for the mugs, and worse performance for the applelogos where compared to [18, 57] [1] . Our performance is comparable to [41] at the operating point selected by them.

Figure 8b characterizes the quality of segmentation by measuring coverage (in blue) and precision (in red) as a function of recall, following [18]: for each correctly detected object instance, coverage is measured as the percentage of points in the ground truth contours that are closer than a threshold t (4% the ground truth bounding box diagonal) from the segmented shape. Precision is measured as the percentage of points of the segmented shape that are closer than t from the ground truth contours. The algorithm clearly has a stable performance in both coverage ($\sim$90%) and accuracy ($\sim$80%), and improves on [18] (data from other references have not been available) in three categories, but not in the categories of applelogos and mugs, probably because we did not represent the leaf of the logo and the full handle of the mug.

7 Conclusion

We have presented a novel generative shape language that can *(i)* effectively model significant category shape variation by a few prototypes and a few parameters for each prototype, and *(ii)* lends itself to efficient search via dynamic programming. We have developed an approach for finding multiple instances of a category in an image, and defined a more sensitive cost function to measure image support for each hypothesis. This top-down approach complements the bottom-up approach of [39] in that both use shape fragments as the basic feature to describe objects. Our recognition results are state of the art on the ETHZ dataset and the segmentation results are excellent. We plan on a better characterization of the shape fragments in the modeling stage, incorporating appearance in the fragment cost functions and using contour fragments instead of edges, which are expected to improve on our preliminary choices.

[1] That our performance in the apple logo and mug categories was not as good as the other three categories was probably because not enough prototype scales were used. In a pilot experiment where more scales were used for larger images, *i.e.*, the number of prototype scales was set proportional to the image size, with a multiplication step size of $\sqrt{2}$, our object-detection performance on the applelogos category exceeded that of both [18, 57].

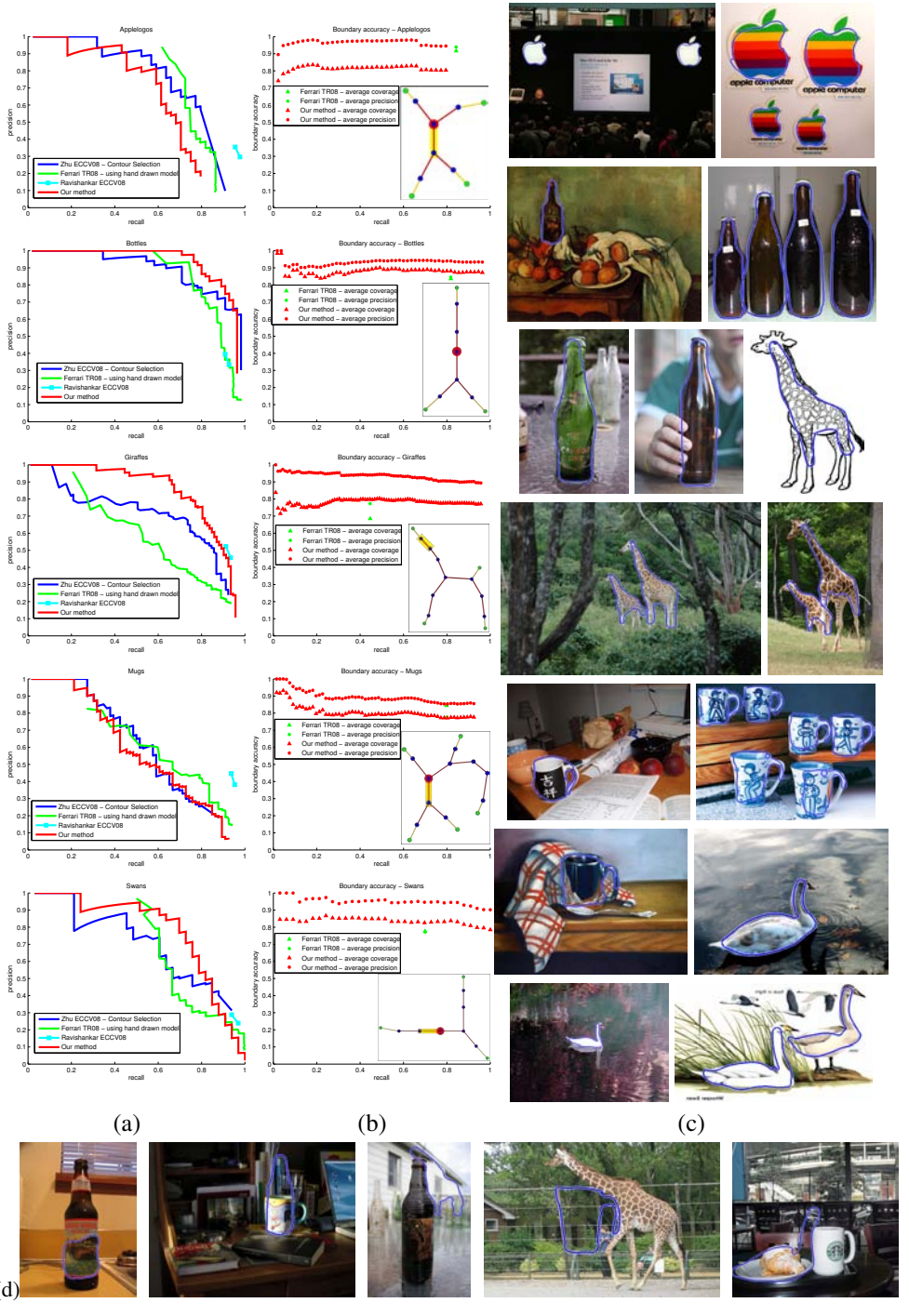


Figure 8: Comparisons of (a) precision-recall and (b) average boundary accuracy for the five categories in the ETHZ dataset. (c) Example of segmentations of the ETHZ dataset. (d) Examples of false positive detections.

References

[1] Yali Amit and Augustine Kong. Graphical templates for model registration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(3):225–236, 1996.

[2] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[3] Alexander C. Berg, Tamara L. Berg, and Jitendra Malik. Shape matching and object recognition using low distortion correspondences. In *CVPR'05*, pages 26–33. IEEE Computer Society, 2005.

[4] Umberto Bertele and Francesco Brioschi. *Nonserial Dynamic Programming*. Academic Press, Inc., Orlando, FL, USA, 1972. ISBN 0120934507.

[5] Haili Chui and Anand Rangarajan. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.*, 89(2-3):114–141, 2003. ISSN 1077-3142. doi: http://dx.doi.org/10.1016/S1077-3142(03)00009-2.

[6] James Coughlan, Alan Yuille, Camper English, and Dan Snow. Efficient deformable template detection and localization without user initialization. *Comput. Vis. Image Underst.*, 78(3):303–319, 2000. ISSN 1077-3142. doi: http://dx.doi.org/10.1006/cviu.2000.0842.

[7] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C Bray. Visual categorization with bags of keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision*, 2004.

[8] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR'05*, pages 886–893. IEEE Computer Society, 2005.

[9] D.Martin, C. Fowlkes, and J.Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):530–549, 2004.

[10] Gyuri Dorkó and Cordelia Schmid. Selection of scale-invariant parts for object class recognition. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, pages 634–640. IEEE Computer Society, 2003. ISBN 0-7695-1950-4.

[11] Pedro F. Felzenszwalb. Representation and detection of deformable shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2):208–220, 2005.

[12] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[13] Pedro F. Felzenszwalb and Joshua D. Schwartz. Hierarchical matching of deformable shapes. In *CVPR'07*. IEEE Computer Society, 2007.

[14] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 264–271, Madison, Wisconsin, June 16-22 2003. IEEE Computer Society Press. URL citeseer.nj.nec.com/580536.html.

[15] Vittorio Ferrari, Tinne Tuytelaars, and Luc Van Gool. Object detection by contour segment networks. In *ECCV'06*, volume 3951 of *Lecture Notes in Computer Science*, pages 14–28. Springer, 2006.

[16] Vittorio Ferrari, Frederic Jurie, and Cordelia Schmid. Accurate object detection with deformable shape models learnt from images. In *CVPR'07*, pages 1–8. IEEE Computer Society, 2007.

[17] Vittorio Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(1):36–51, 2008.

[18] Vittorio Ferrari, L. Fevrier, F. Jurie, and C. Schmid. From images to shape models for object detection. Technical Report 6600, INRIA, 2008.

[19] Stuart Geman and Kevin Kochanek. Dynamic programming and the graphical representation of error-correcting codes. *IEEE Transactions on Information Theory*, 47(2):549–568, 2001.

[20] Peter J. Giblin and Benjamin B. Kimia. On the intrinsic reconstruction of shape from its symmetries. *PAMI*, 25(7):895–911, July 2003.

[21] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the Hausdorff distance. *PAMI*, 15:850–863, 1993.

[22] Vishal Jain, Benjamin B. Kimia, and Joseph L. Mundy. Segregation of moving objects using elastic matching. *Computer Vision and Image Understanding*, 108:230–242, 2007.

[23] F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *CVPR*, pages II: 90–96, 2004.

[24] Frederic Jurie and Bill Triggs. Creating efficient codebooks for visual recognition. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 604–610. IEEE Computer Society, 2005. ISBN 0-7695-2334-X-01.

[25] Michael F. Kelly and Martin D. Levine. Annular symmetry operators: A method for locating and describing objects. In *iccv*, 1995.

[26] Benjamin B. Kimia. On the role of medial geometry in human vision. *Journal of Physiology-Paris*, 97(2–3):155–190, 2003.

[27] P. D. Kovesi. MATLAB and Octave functions for computer vision and image processing. School of Computer Science & Software Engineering, The University of Western Australia, 2009. Available from: <http://www.csse.uwa.edu.au/∼pk/research/matlabfns/>.

[28] M. Pawan Kumar, Philip H. S. Torr, and A. Zisserman. Obj cut. In *CVPR'05*, pages 18–25. IEEE Computer Society, 2005.

[29] Y. Lamdan, JT Schwartz, and HJ Wolfson. Affine invariant model-based object recognition. *Robotics and Automation, IEEE Transactions on*, 6(5):578–589, 1990.

[30] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR'06*, pages 2169–2178. IEEE Computer Society, 2006.

[31] Bastian Leibe and Bernt Schiele. Scale-invariant object categorization using a scale-adaptive mean-shift search. In *DAGM-Symposium*, pages 145–153, 2004.

[32] Marius Leordeanu, Martial Hebert, and Rahul Sukthankar. Beyond local appearance: Category recognition from pairwise interactions of simple features. In *CVPR'07*. IEEE Computer Society, 2007.

[33] L. Lin, S.W. Peng, J. Porway, S.C. Zhu, and Y.T. Wang. An empirical study of object category recognition: Sequential testing with generalized samples. In *ICCV07*, pages 1–8, 2007.

[34] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[35] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, 2005.

[36] Andreas Opelt, Axel Pinz, and Andrew Zisserman. Incremental learning of object detectors using a visual shape alphabet. In *CVPR'06*, pages 3–10. IEEE Computer Society, 2006.

[37] Andreas Opelt, Axel Pinz, and Andrew Zisserman. A boundary-fragment-model for object detection. In *ECCV'06*, volume 3951 of *Lecture Notes in Computer Science*, pages 575–588. Springer, 2006.

[38] Andreas Opelt, Axel Pinz, and Andrew Zisserman. Learning an alphabet of shape and appearance for multi-class object detection. *International Journal of Computer Vision*, 80(1):16–44, 2008.

[39] Ozge C. Ozcanli and Benjamin B. Kimia. Generic object recognition via shock patch fragments. In Nasir M. Rajpoot and Abhir Bhalerao, editors, *Proceedings of the British Machine Vision Conference*, pages 1030–1039, Coventry, USA, September 10-13 2007. Warwick Print.

[40] Ozge C. Ozcanli, Amir Tamrakar, Benjamin B. Kimia, and Joseph L. Mundy. Augmenting shape with appearance in vehicle category recognition. In *CVPR'06*, pages 935–942. IEEE Computer Society, 2006.

[41] Saiprasad Ravishankar, Arpit Jain, and Anurag Mittal. Multi-stage contour based detection of deformable objects. In *ECCV (2)*, volume 5303 of *Lecture Notes in Computer Science*, pages 483–496. Springer, 2008. ISBN 978-3-540-88685-3.

[42] Thomas Sebastian, Philip Klein, and Benjamin Kimia. On aligning curves. *PAMI*, 25(1):116–125, January 2003.

[43] Thomas Sebastian, Philip Klein, and Benjamin Kimia. Recognition of shapes by editing their shock graphs. *PAMI*, 26:551–571, May 2004.

[44] Andrea Selinger and Randal C. Nelson. A perceptual grouping hierarchy for appearance-based 3d object recognition. *Computer Vision and Image Understanding*, 76(1):83–92, 1999.

[45] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *iccv*, pages 281–288, 2005.

[46] J. Shotton, A. Blake, and R. Cipolla. Multiscale categorical object recognition using contour fragments. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(7):1270–1281, 2008.

[47] Jamie Shotton, John M. Winn, Carsten Rother, and Antonio Criminisi. *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV'06*, volume 3951 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2006.

[48] Kaleem Siddiqi and Benjamin B. Kimia. Parts of visual form: Computational aspects. *PAMI*, 17 (3):239–251, March 1995.

[49] Kaleem Siddiqi, Kathryn J. Tresness, and Benjamin B. Kimia. Parts of visual form: Ecological and psychophysical aspects. *Perception*, 25:399–424, 1996.

[50] Kaleem Siddiqi, Benjamin B. Kimia, Allen R. Tannenbaunm, and Steven W. Zucker. On the psychophysics of the shape triangle. *Vision Research*, 41(9):1153–1178, 2001.

[51] Huseyin Tek and Benjamin B. Kimia. Symmetry maps of free-form curve segments via wave propagation. *IJCV*, 54(Issue 1-3):35–81, August 2003.

[52] Sinisa Todorovic and Narendra Ahuja. Extracting subimages of an unknown category from a set of images. In *CVPR'06*, pages 927–934. IEEE Computer Society, 2006.

[53] A. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR'04*, pages 762–769. IEEE Computer Society, 2004.

[54] Nhon H. Trinh and Benjamin B. Kimia. A symmetry-based generative model for shape. In *ICCV '07: Proceedings of the Eleventh IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007. IEEE Computer Society.

[55] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 511–518, Kauai, Hawaii, USA, December 9-14 2001. IEEE Computer Society Press.

[56] John M. Winn and Nebojsa Jojic. Locus: Learning object classes with unsupervised segmentation. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*, pages 756–763. IEEE Computer Society, 2005. ISBN 0-7695-2334-X.

[57] Qihui Zhu, Liming Wang, Yang Wu, and Jianbo Shi. Contour context selection for object detection: A set-to-set contour matching approach. In *ECCV*, volume 5303 of *Lecture Notes in Computer Science*, pages 774–787. Springer, 2008. ISBN 978-3-540-88685-3.