Shape-based Object Discovery in Images

Sinisa Todorovic and Nadia Payet

Oregon State University, Corvallis, OR 97331, USA
sinisa@eecs.oregonstate.edu

Abstract. This paper presents an overview of our recent work on shape-based object recognition in images. The overview focuses on the following related problems: i) discovery of all distinct 2D object categories frequently occurring in an unlabeled set of images; ii) learning a model of the discovered categories; and iii) recognition and localization of objects from the discovered categories in new images. The paper argues that using image contours as basic features, and thus directly grounding object discovery and recognition on shape, offers a number of advantages in solving (i)-(iii) over more commonly used point features. Since shape is directly encoded by layouts of image contours, similar contour layouts across the images are expected to belong rather to object occurrences, than the background. The contour layouts are captured by a graph over all pairs of matching contours from different images. The graph's maximum a posteriori multi-coloring assignment is taken to represent the shapes of discovered objects. Our empirical evaluation suggests that shape is more expressive and discriminative than photometric features for object discovery.

1 Introduction

This paper presents an overview of the shape-based approach to object recognition and related problems that we have developed over the last several years [1–3]. We briefly describe the major components of our work, and explain its advantages over the more common methods based on point features (e.g., [4–11]).

The role of shape in representing and recognizing objects in images is a long-standing question in computer vision. In psychophysics, it is widely recognized that shape is one of the most categorical object properties [12]. Yet, most recent work on object recognition exclusively resorts to appearance features (e.g., color, textured patches), arguing that they are more stable to variations in imaging conditions (e.g., illumination, viewpoint). However, there are a number of unsatisfying aspects associated with point features. They are usually defined only in terms of local discontinuities in brightness. The inherent locality of points cannot represent the full spatial extent of objects in the image. As a direct consequence, point-based object detection requires the use of scanning windows of pre-specified size and shape, resulting in overlapping candidate detections that need to be resolved in a postprocessing step (e.g., non-maxima suppression). This postprocessing is usually based on heuristic assumptions about the numbers, sizes, and shapes of objects present. Since the final result of this is identification of the points associated with detected objects, it leads to only approximate object localization.

A number of approaches, including our previous work, use image contours as features [11, 13–25]. These methods argue that contours are in general richer descriptors, more discriminative, and more noise-tolerant than interest points. Contours make various constraints, frequently used in object recognition—such as those dealing with continuation, smoothness, containment, and adjacency—implicit and easier to incorporate than points. Contours often coincide with the boundaries of objects and their subparts. This allows simultaneous object detection and segmentation. Shape-based recognition typically requires a manually specified shape template [21, 22], or manually segmented training images to learn the object shape [26]. Such a high level of supervision in training can be relaxed by combining shape with point features [27, 28].

It is worth noting that the impact of any shortcomings of a contour detection algorithm should not be confused with the weaknesses of shape-based representation. For example, oversimplifying assumptions made by some edge detection algorithms about shape, curvature, size, gray-level contrast, and topological context of objects to be expected in an image may lead to various errors [29–31]. From our experience, these errors could be addressed by a higher-level recognition algorithms, as presented here.

In this paper, we study the role of object shape in the problem of discovering instances of frequently occurring object categories (e.g, faces, bikes, giraffes, etc.) in an unlabeled set of images. Object discovery is arguably a more difficult problem than learning visual properties of objects from labeled images, since the former additionally requires identifying a meaningful image content in the background clutter, whereas the latter exploits human annotation for directly accessing the image content of interest. Object discovery brings together most recognition related problems of interest here, and serves well to highlight the strengths and shortcomings of using shape as object features for recognition. In particular, for object discovery, we deliberately disregard appearance features, and use only the geometric properties of image contours. In this way, we are in a position to empirically evaluate if shape is expressive and discriminative enough to provide robust detection and segmentation of common objects in the midst of background clutter. Also, we can empirically show advantages of using only shape-based cues over photometric features for object discovery.

Most previous work on unsupervised object discovery exploits photometric properties of objects. For example, color of image regions is used in [32, 33], and texture properties of image patches are used in [34, 35]. In our experiments, we outperform these appearance-based approaches to object discovery in both object detection and segmentation on benchmark datasets.

The remainder of this paper is organized as follows. Sec. 2 briefly reviews our approach to object discovery and points out our contributions. Sec. 3 specifies our shape representation. Sec. 4 describes how to build a graph from all pairs of image contours to capture shape properties of objects. Sec. 5 presents our graph multicoloring algorithm for object discovery. Sec. 5 presents our experimental evaluation. Finally, Sec. 7 presents our concluding remarks.

2 A Brief Review of Our Approach

This section reviews our approach, originally presented in [2]. It consists of three steps, illustrated in Fig. 1. **Step 1:** Given a set of unlabeled images, we extract their contours by the minimum-cover algorithm of [36]. Each contour is characterized as a sequence

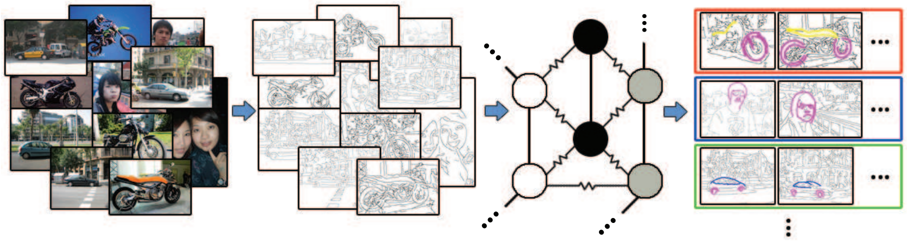


Fig. 1. Overview: Given a set of unlabeled images (left), we extract their contours (middle left), and then build a graph of pairs of matching contours. Contour pairs are viewed as collaborating (straight graph edges), if they similarly deform from one image to another, or conflicting (zigzag graph edges), otherwise. Such coupling of contour pairs facilitates their clustering with the Coordinate Ascent Swendsen-Wang cut (CASW). The resulting clusters represent shapes of discovered objects (right). (best viewed in color)

of beam-angle histograms, computed at points sampled along the contour. Similarity between two contours is estimated by the dynamic time warping (DTW) of the corresponding sequences of beam-angle descriptors. **Step 2** builds a weighted graph of matching contours, aimed at facilitating a separation of the background from object shapes in Step 3. We expect that there will be many similarly shaped curves, belonging to the background. Since the backgrounds vary, by definition, similar background curves will most likely have different spatial layouts across the image set. In contrast, object contours (e.g., curves delineating a giraffe's neck) are more likely to preserve both shape and layout similarity in the set. Therefore, for object discovery, it is critical that we capture similar configurations of contours. We build a graph, where nodes correspond to pairs of matching contours, and graph edges capture spatial layouts of quadruples of contours. **Step 3** conducts a probabilistic, iterative multicoloring of the graph using the Coordinate-Ascent Swendsen-Wang (CASW) cut. In each iteration, CASW cut probabilistically samples graph edges, and then assigns colors to the resulting groups of connected nodes. The assignments are accepted by the Metropolis-Hastings (MH) mechanism. After convergence, the resulting clusters represent shapes of objects that are discovered in the image set.

3 Image Representation Using Shapes and Shape Description

This section presents Step 1 of our approach. In each image, we extract relatively long, open contours using the minimum-cover algorithm of [36], referred to as gPb+ [36]. Similarity between two contours is estimated by aligning their sequences of points by the Dynamic Time Warping (DTW). Each contour point is characterized by the weighted Beam Angle Histogram (BAH), illustrated in Fig. 2. BAH is a weighted version of the standard unweighted BAH, aimed at mitigating the uncertainty in contour extraction. BAH down-weights the interaction of distant contour parts, as they are more likely to belong to distinct objects in the scene, rather than to the same objects. BAH is invariant to translation, in-plane rotation, and scale. Experimentally, we find that BAH

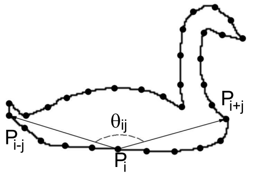


Fig. 2. BAH is a weighted histogram of beam angles θ_{ij} at contour points P_i, i=$1, 2, \ldots$

Contour detectors	**BAH**	BAH-U	[37]	[38]	[27]
Canny	**0.23±0.01**	0.21	0.18	0.15	0.21
	0.59±0.02	0.57	0.48	0.48	0.52
[28]	**0.32±0.03**	0.30	0.25	0.18	0.29
	0.78±0.03	0.75	0.62	0.61	0.72
gPb+ [36]	**0.37±0.02**	0.34	0.26	0.20	0.34
	0.81±0.03	0.78	0.63	0.61	0.74

Table 1. Contour matching on the ETHZ image dataset [28]. Top is $Precision$, bottom is $Recall$. The rightmost column shows matching results of Oriented Chamfer Distance [27], and other columns show DTW results. Descriptors (left to right): our BAH, unweighted BAH, Shape Context [37], and SIFT [38].

with 12 bins gives optimal and stable results, and seems more robust to errors in contour extraction than some alternative shape descriptors, as reported in Table 1.

4 Constructing the Graph of Pairs of Image Contours

This section presents Step 2 that constructs a weighted graph, $G = (V, E, \rho)$, from contours extracted from all images in the set. Nodes of G represent candidate matches of contours, $(u, u')\in V$, where u and u' belong to two different images. Similarity of two contours is estimated by DTW. We keep only the best 5% of contour matches as nodes of G. The graph is instrumental in capturing both intrinsic geometric properties of shape parts, and relative layout relationships between shape parts. This facilitates generating hypotheses of frequently occurring objects in the image set as similar contours repeating in similar layouts in the images.

Edges of G, $e = ((u, u'), (v, v')) \in E$, capture spatial relations of corresponding image contours. If contours u and v in image 1, and their matches u' and v' in image 2 have similar spatial layout, then they are less likely to belong to the background clutter. All such contour pairs will have a high probability to become positively coupled in G. Otherwise, matches (u, u') and (v, v') will have a high probability to become negatively coupled in G, so that they could be placed in distinct clusters. This probabilistic coupling of nodes in G is encoded by edge weights, ρ_e, defined as the likelihood $\rho_e^+ \propto \exp(-w_\delta^+ \delta_e)$, given the positive polarity of e, and $\rho_e^- \propto \exp(-w_\delta^- (1-\delta_e))$, given the negative polarity of e. w_δ^+ and w_δ^- are the parameters of the exponential distribution, and $\delta_e \in [0, 1]$ measures a difference in spatial layouts of u and v in image 1, and their matches u' and v' in image 2.

We specify δ_e so as to account for small object pose and camera viewpoint differences across the images. From our experiments, this is critical for enabling robustness in the face of noise in contour extraction and representation. We make a distinction between the following two cases.

Case 1: (u, u') and (v, v') come from *two* images, where u and v are in image 1, and u' and v' are in image 2, as illustrated in Fig. 3. We estimate δ_e in terms of affine homographies between the matching contours, denoted as $H_{uu'}$, and $H_{vv'}$, as follows.

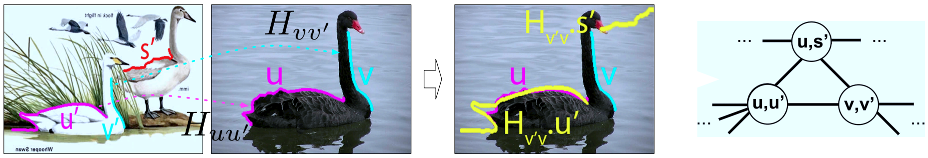


image 1 image 2 homographic projection association graph

Fig. 3. Estimating layout difference $\delta_{(u,u',v,v')}$ when contours u and v are in image 1, and their matches u' and v' are in image 2. We use the affine-homography projection of u' and v' to image 1, $u'' = H_{vv'}u'$ and $v'' = H_{uu'}v'$, and compute δ as the average distance between u and u'', and v and v''. The figure with projections shows that the contours (u, s', v, v') have different layouts in image 1 and image 2, whereas the contours (u, u', v, v') have a similar layout.

From the DTW alignment of points along u and u', we estimate their affine homography $H_{uu'}$. Similarly, for v and v', we estimate $H_{vv'}$. Then, we project u' to image 1, as $u''{=}H_{vv'}u'$, and, similarly, project v' to image 1 as $v''{=}H_{uu'}v'$ (Fig. 3 right). Next, in image 1, we measure distances between corresponding points of u and u'', where the point correspondence is obtained from DTW of u and u'. Similarly, we measure distances between corresponding points of v and v''. δ_e is defined as the average point distance between u and u'', and v and v''.

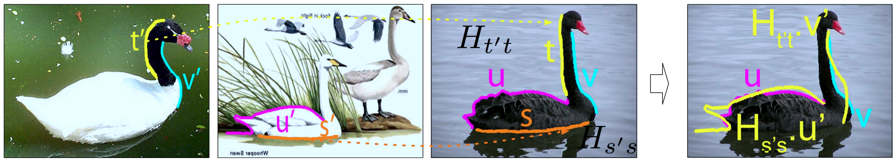


image 1 image 2 image 3 homographic projection

Fig. 4. Estimating layout difference $\delta_{(u,u',v,v')}$ when contours u and v are in image 1, and their matches u' and v' are in image 2 and image 3, respectively. We use auxiliary contours s in the neighborhood of u to estimate multiple affine-homography projections of u' to image 1, $u''_s = H_{ss'}u'$, where s' is the best matching contour of s in image 2. Also, we use auxiliary contours t in the neighborhood of v to estimate multiple projection of v' to image 1, $v''_t = \sum_s H_{tt'}v'$, where t' is the best matching contour of t in image 3. On the right, we show example projections $u''_s = H_{ss'}u'$ and $v''_t = H_{tt'}v'$. Finally, we compute δ as the average distance between u and $\{u''_s\}$, and v and $\{v''_t\}$.

Case 2: (u, u') and (v, v') come from *three* images, where u and v belong to image 1, u' is in image 2, and v' is in image 3, as illustrated in Fig. 4. In this case, we can neither use $H_{vv'}$ to project u' from image 2 to image 1, nor $H_{uu'}$ to project v' from image 3 to image 1. Instead, we resort to context provided by auxiliary contours s' in a vicinity of u', and auxiliary contours t' in a vicinity of v'. For every neighbor s' of u' in image 2, we find its best DTW match s in image 1, and compute homography $H_{ss'}$. Similarly, for every neighbor t' of v' in image 3, we find its best DTW match t in image 1, and compute homography $H_{tt'}$. Then, we use all these homographies to project u' to image 1, multiple times, as $u''_s{=}H_{ss'}u'$, for each neighboring contour s. Similarly, we project v'' to image 1, multiple times, as $v''_t{=}H_{tt'}v'$, for each neighboring contour t. Next, as in Case 1, we measure distances between corresponding points of all u and $\{u''_s\}$ pairs, and all v and $\{v''_t\}$ pairs. δ_e is defined as the average point distance.

5 Coordinate-Ascent Swendsen-Wang Cut

This section presents Step 3. Our goal is to perform multicoloring of the graph of contour matches, $G = (V, E, \rho)$, specified in the previous section. The multicoloring partitions G into two subgraphs. One subgraph will represent a composite cluster of nodes, consisting of a number of connected components (CCPs), receiving distinct colors. This composite cluster contains contours of the discovered object categories. Nodes outside of the composite cluster are interpreted as the background. An edge, $e \in E$, can be negative or positive. A negative edge indicates that the nodes are conflicting, and thus should not be assigned the same color. A positive edge indicates that the nodes are collaborative, and thus should be favored to get the same color. If nodes are connected by positive edges, they form a CCP, and receive the same color. A CCP cannot contain a negative edge. CCPs connected by negative edges form a composite cluster. The amount of conflict and collaboration between two nodes is defined by the likelihood ρ, defined in Sec. 4.

For multicoloring of G, we use the Coordinate Ascent Swendsen-Wang cut (CASW) that iterates the following three steps: (1) Sample a composite cluster from G, by probabilistically cutting and sampling positive and negative edges between nodes of G. This results in splitting and merging nodes into a new configuration of CCPs. (2) Assign new colors to the resulting CCPs within the selected composite cluster, and use the Metropolis-Hastings (MH) algorithm [39] to estimate whether to accept this new multicoloring assignment of G, or to keep the previous state. (3) If the new state is accepted, go to step (1); otherwise, if the algorithm converged, re-estimate parameters of the pdf's controlling the MH iterations, and go to step (1), until the pdf re-estimation does not affect convergence. CASW is characterized by large MH moves, involving many strongly-coupled graph nodes. This typically helps avoid local minima, and allows fast convergence, unlike other related MCMC methods (e.g., [40]). In the following, we present our Bayesian formulation of the CASW cut.

5.1 Bayesian Formulation

Multi-coloring of G amounts to associating labels l_i to nodes in V, $i{=}1, \dots, |V|$, where $l_i \in \{0, 1, \dots, K\}$. K denotes the total number of target objects, which is a priori unknown, and $(K + 1)$th label is the background. The multicoloring can be formalized as $\mathcal{M}{=}(K, \{l_i\}_{i=1,\dots,|V|})$. To find $\mathcal{M}$, we maximize the posterior $p(\mathcal{M}|G)$, as

$$\mathcal{M}^* = \arg\max_{\mathcal{M}} p(\mathcal{M}|G) = \arg\max_{\mathcal{M}} p(\mathcal{M})p(G|\mathcal{M}). \tag{1}$$

We define the prior as $p(\mathcal{M}) \propto \exp(-w_K K)\exp(-w_N N)$, where N is the number of nodes that are labeled as background, and w_K and w_N are the parameters of the exponential distribution. $p(\mathcal{M})$ penalizes large K and N.

We specify the likelihood, $p(G|\mathcal{M})$, in terms of independent Bernoulli edges of G. We define binary functions $\mathbb{1}_{l_i \neq l_j}$ and $\mathbb{1}_{l_i = l_j}$, which indicate whether node labels l_i and l_j are different, or the same. Then, we have

$$p(G|\mathcal{M}) \propto \textstyle\prod_{e\in\mathbb{E}^+} \rho_e^+ \prod_{e\in\mathbb{E}^-} \rho_e^- \prod_{e\in\mathbb{E}^0} (1-\rho_e^+)\mathbb{1}_{l_i\neq l_j} \cdot (1-\rho_e^-)\mathbb{1}_{l_i=l_j}\ , \qquad (2)$$

where $\mathbb{E}^+$ and $\mathbb{E}^-$ are the sets of positive and negative edges present in the composite cluster, and $\mathbb{E}^0$ is the set of edges that are probabilistically cut.

5.2 Inference Using the CASW Cut

The CASW cut iterates the following two steps in inference. In step (1), edges of G are probabilistically sampled. If two nodes have the same label, their positive edge is sampled, with likelihood ρ_e^+. Otherwise, if the nodes have different labels, their negative edge is sampled, with likelihood ρ_e^-. This re-connects all nodes into new connected components (CCPs). The negative edges that are sampled will connect CCPs into a number of composite clusters, denoted by V_{cc}. This configuration is referred to state A. In step (2), we choose at random one composite cluster, V_{cc}, and probabilistically reassign new colors to the CCPs within V_{cc}, resulting in a new state B.

The CASW accepts the new state B as follows. Let $q(A \to B)$ be the proposal probability for moving from state A to B, and let $q(B \to A)$ denote the reverse. The acceptance rate, $\alpha(A{\to}B)$, of the move from A to B is defined as

$$\alpha(A \to B) = \min\left(1, \frac{q(B \to A)p(\mathcal{M}=\mathcal{B}|\mathcal{G})}{q(A \to B)p(\mathcal{M}=\mathcal{A}|\mathcal{G})}\right). \qquad (3)$$

If $\alpha(A \to B)$ is low, state B cannot be accepted, and CASW remains in state A.

$q(A \to B)$ is defined as a product of two probabilities: (i) the probability of generating V_{cc} in state A, $q(V_{cc}|A)$; and (ii) the probability of recoloring the CCPs within V_{cc} in state B, where V_{cc} is obtained in state A, $q(B(V_{cc})|V_{cc}, A)$. Thus, we have

$$\frac{q(B{\to}A)}{q(A{\to}B)} = \frac{q(V_{cc}|B)}{q(V_{cc}|A)} = \frac{\prod_{e\in \text{Cut}_B^+}(1{-}\rho_e^+)\prod_{e\in \text{Cut}_B^-}(1{-}\rho_e^-)}{\prod_{e\in \text{Cut}_A^+}(1{-}\rho_e^+)\prod_{e\in \text{Cut}_A^-}(1{-}\rho_e^-)}\ . \qquad (4)$$

Note that complexity of each move is relatively low, since computing $\frac{q(B\to A)}{q(A\to B)}$ involves only those edges that are probabilistically cut around V_{cc} in states A and B — not all edges. Also, $\frac{p(\mathcal{M}=\mathcal{B}|\mathcal{G})}{p(\mathcal{M}=\mathcal{A}|\mathcal{G})} = \frac{p(\mathcal{M}=\mathcal{B})p(G|\mathcal{M}=B)}{p(\mathcal{M}=\mathcal{A})p(G|\mathcal{M}=A)}$ can be efficiently computed. $p(\mathcal{M}=\mathcal{B})$ can be directly computed from the new coloring in state B, and $\frac{p(G|\mathcal{M}=B)}{p(G|\mathcal{M}=A)}$ depends only on those edges that have changed their polarity.

6 Results

This section reviews the empirical validation of our approach, presented in [2]. The experiments demonstrate advantages of using shape-based representations and modeling of objects for recognition versus alternative approaches.

Caltech categories	**Our method**	[35]	[34]	[41]
A,C,F,M	**98.62±0.51**	98.03	98.55	88.82
A,C,F,M,W	**97.57±0.46**	96.92	97.30	N/A
A,C,F,M,W,K	**97.13±0.42**	96.15	95.42	N/A

ETHZ categories	**Our method**	[35]
A,B,G,M,S (bbox)	**96.16±0.41**	95.85
A,B,G,M,S (expanded)	**87.35±0.37**	76.47
A,B,G,M,S (entire image)	**85.49±0.33**	N/A

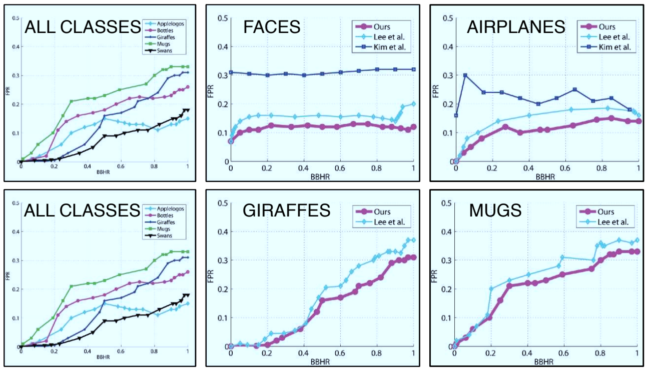
Table 2. Mean purity of category discovery for Caltech-101 (A:Airplanes, C: Cars, F: Faces, M: Motorbikes, W: Watches, K: Ketches), and ETHZ dataset (A:Applelogos, B: Bottles, G: Giraffes, M: Mugs, S: Swans).

Given a set of images, we perform object discovery in two stages, as in [34, 35, 41]. We first coarsely cluster images based on their contours using CASW cut, and then again use CASW to cluster contours from only those images that belong to the same coarse cluster. The first stage serves to discover different object categories in the image set. The second, fine-resolution stage serves to separate object contours from the background, and identify characteristic parts of each discovered object category.

We use the following benchmark datasets: Caltech-101 [42], ETHZ [28], LabelMe [43], and Weizmann Horses [44]. In the experiments on Caltech-101, we use all Caltech images showing the same categories as those used in [34]. Evaluation on ETHZ and Weizmann Horses uses the entire datasets. For LabelMe, we keep the 15 first images retrieved by keywords *car side*, *car rear*, *face*, *airplane* and *motorbike*. ETHZ and LabelMe increase complexity over Caltech-101, since their images contain multiple object instances, which may: (a) appear at different resolutions, (b) have low contrasts with textured background, and (c) be partially occluded. The Weizmann Horses are suitable to evaluate performance on articulated, non-rigid objects.

In the first stage of object discovery, CASW finds clusters of images. This is evaluated by *purity*. Purity measures the extent to which a cluster contains images of a single dominant object category. In the second stage, on each of these image clusters, we use *Bounding Box Hit Rate* (BBHR) to verify whether contours detected by CASW fall within the true foreground regions. The ground truth is defined as all pixels of the extracted image contours that fall in the bounding boxes or segments of target objects. A contour detected by CASW is counted as "hit" whenever the contour covers 50% or more of the ground-truth pixels. Since we discard contours that are less than 50 pixels, this means that at least 25 ground-truth pixels need to be detected within the bounding box. Our accuracy in the second clustering stage depends on the initial set of pairs of matching contours (i.e., nodes of graph G) input to CASW. This is evaluated by plotting the ROC curve, parameterized by a threshold on the minimum DTW similarity between pairs of matching contours which are included in G.

We evaluate the first and second stages of object discovery. *First Stage:* We build a weighted graph whose nodes represent entire images. Edges between images in the graph are characterized by weights, defined as an average of DTW similarities of contour matches from the corresponding pair of images. A similar characterization of graph edges is used in [34,35]. For object discovery, we apply CASW to the graph, resulting in image clusters. Each cluster is taken to consist of images showing a unique object category. Unlike [34, 35], we do not have to specify the number of categories present in the image set, as an input parameter, since it is automatically inferred by CASW. Evaluation is done on Caltech-101 and the ETHZ dataset. Table 2 shows that our mean purity is



	CASW	[34]	[35]
A	**0.11±0.01**	0.21	0.17
F	**0.12±0.01**	0.30	0.15
K	**0.06±0.003**	0.19	0.08
M	**0.04±0.002**	0.11	0.07
W	**0.02±0.003**	0.08	0.03

	CASW	[34]	[35]
A	**0.15±0.02**	N/A	0.18
B	**0.18±0.01**	N/A	0.20
G	**0.16±0.01**	0.32	0.18
M	**0.23±0.04**	N/A	0.27
S	**0.09±0.002**	N/A	0.11

Fig. 5. Bounding Box Hit Rates (BBHR) vs False Positive Rates (FPR). Top is Caltech-101, bottom is ETHZ. Left column is our CASW on all classes, and middle and right columns show a comparison with [34, 35] on a specific class (lower curves are better). The tables show FPR at BBHR=0.5. Caltech-101: A: Airplanes, F: Faces, K: Ketches, M: Motorbikes, W: Watches. ETHZ: A: Applelogs, B: Bottles, G: Giraffes, M: Mugs, S: Swans. (best viewed in color)

superior to that of [34, 35, 41]. On Caltech-101, CASW successively finds $K = 4, 5, 6$ clusters of images, as we gradually increase the true number of categories from 4 to 6. This demonstrates that we are able to automatically find the number of categories present, with no supervision. On ETHZ, CASW again correctly finds $K = 5$ categories. As in [35], we evaluate purity when similarity between the images (i.e., weights of edges in the graph) is estimated based on contours falling within: (a) the bounding boxes of target objects, (b) twice the size of the original bounding boxes (called expanded in Table 2), and (c) the entire images. On ETHZ, CASW does not suffer a major performance degradation when moving from the bounding boxes, to the challenging case of using all contours from the entire images. Overall, our purity rates are high, which enables accurate clustering of contours in the second stage. *Second Stage:* We use contours from all images grouped within one cluster, found in the first stage, to build our graph G, and then conduct CASW. This is repeated for all image clusters. The clustering of contours by CASW amounts to foreground detection, since the identified contour clusters are taken to represent parts of the discovered object category. We evaluate BBHR and FPR on Caltech-101, ETHZ, LabelMe, and Weizmann Horses. Fig.5 shows that our BBHR and FPR values are higher than those of [34, 35] on the Caltech and ETHZ. CASW finds $K = 1$ for *Airplanes, Cars Rear, Faces, Ketches, Watches* in Caltech-101, *Apples, Bottles, Mugs* in ETHZ, and *Car rear, Face, Airplane* in LabelMe. These objects do not have articulated parts that move independently, hence, only one contour cluster is found. On the other hand, it finds $K = 2$ for *Giraffes, Swans* in ETHZ, *Cars side, Motorbikes* in Caltech and LabelMe, and $K = 3$ for Weizmann Horses. In Fig.6, we highlight contours from different clusters with distinct colors. Fig.6 demonstrates that CASW is capable not only to discover foreground objects, but also to detect their characteristic parts, e.g., wheels and roof for *Cars side*, wheels and seat for *Motorbikes*, head and legs for *Giraffes*, etc. The plot in Fig.6 evaluates our object detection on LabelMe and Weizmann Horses. Detection accuracy is estimated as the standard ratio of intersection over union of ground-truth and detection bounding boxes, $(BB_{gt} \cap BB_d)/(BB_{gt} \cup BB_d)$, where BB_d is the smallest bounding box that encloses detected contours in the image. The average detection accuracy for each category is: [Face(F): 0.52, Airplane(A): 0.45, Motorbike(M): 0.42, Car Rear(C): 0.34], whereas [35] achieves only [(F): 0.48, (A): 0.43, (M): 0.38, (C): 0.31]. For Weizmann Horses, we obtain $Precision$ and $Recall$ of 84.9%±0.68% and 82.4%±0.51%, whereas [33] achieves only 81.5% and 78.6%.

The C-implementation of our CASW runs in less than 2 minutes on any dataset of less than 100 images, on a 2.40GHz PC with 3.48GB RAM.

7 Conclusion

We have argued in this paper that using contours as basic image features: (a) Facilitates capturing shape properties of objects; (b) Allows a unified computational framework that can jointly address object discovery, recognition, and segmentation; and (c) Enables efficient and robust learning and inference. Our claims are supported by the state-of-the-art performance of our shape-based approach to object discovery, recognition, and segmentation, which we have reviewed in this paper. Our approach clusters image contours based on their intrinsic geometric properties, and spatial layouts. The resulting clusters are interpreted as shapes of parts of discovered objects.

We have derived two key insights. First, shape alone is sufficiently discriminative and expressive to provide robust and efficient object discovery in unlabeled images, which even outperforms related point-based methods. As image contours are dimensionally matched with shape they are more suitable features for object discovery than point features. Second, due to background clutter, there could be many similar image features — both contours and point features — coinciding with true object occurrences and the background. To separate the background from foreground in object discovery, one usually makes the assumption that the background clutter cannot generate occurrences of similar spatial configurations of features in distinct images with a high probability. This probability is arguably lower for similar spatial configurations of contours than that of points, since contours have a lager spatial extent than points. Thus, identifying similar contour layouts in the images is expected to yield more accurate foreground-background separation than finding similar layouts of points. In summary, using contours facilitates discovering frequently occurring objects in images.

References

1. Payet, N., Todorovic, S.: Matching hierarchies of deformable shapes. In: Proc. 7th IAPR-TC-15 Workshop Graph-based Representations in Pattern Recognition (GbR). (2009) 1–10
2. Payet, N., Todorovic, S.: From a set of shapes to object discovery. In: ECCV. (2010)
3. Payet, N., Todorovic, S.: From contours to 3D object detection and pose estimation. In: ICCV (oral presentation). (2011)
4. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR. Volume 2. (2003) 264–271

5. Torralba, A., Murphy, K., Freeman, W.: Sharing features: efficient boosting procedures for multiclass object detection. In: CVPR. Volume 2. (2004) 762–769
6. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: Workshop on Statistical Learning in Computer Vision, ECCV. (2004) 17–32
7. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. IEEE TPAMI **26**(11) (2004) 1475–1490
8. Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Learning hierarchical models of scenes, objects, and parts. In: ICCV. Volume 2. (2005) 1331–1338
9. Winn, J., Criminisi, A., Minka, T.: Object categorization by learned universal visual dictionary. In: ICCV. Volume 2. (2005) 1800–1807
10. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE TPAMI **28**(4) (2006) 594– 611
11. Opelt, A., Pinz, A., Zisserman, A.: Incremental learning of object detectors using a visual shape alphabet. In: CVPR. Volume 1. (2006) 3–10
12. Biederman, I.: Surface versus edge-based determinants of visual recognition. Cognitive Psychology **20**(1) (January 1988) 38–64
13. Williams, L., Jacobs, D.: Stochastic completion fields: A neural model of illusory contour shape and salience. In: ICCV. (1995) 408–415
14. Lindenbaum, M.: Bounds on shape recognition performance. IEEE Trans. PAMI **17**(7) (1995)
15. Liu, T.L., Geiger, D.: Approximate tree matching and shape similarity. In: ICCV. Volume 1. (1999) 456–462
16. Shokoufandeh, A., Macrini, D., Dickinson, S., Siddiqi, K., Zucker, S.W.: Indexing hierarchical structures using graph spectra. IEEE TPAMI **27**(7) (2005) 1125–1140
17. Keselman, Y., Dickinson, S.: Generic model abstraction from examples. IEEE TPAMI **27**(7) (2005) 1141–1156
18. Siddiqi, K., Kimia, B.B.: A shock grammar for recognition. In: CVPR. (1996) 507
19. Sebastian, T.B., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing their shock graphs. IEEE TPAMI **26**(5) (2004) 550–571
20. Felzenszwalb, P.F.: Representation and detection of deformable shapes. IEEE TPAMI **27**(2) (2005) 208–220
21. Zhu, Q., Wang, L., Wu, Y., Shi, J.: Contour context selection for object detection: A set-to-set contour matching approach. In: ECCV (2). (2008) 774–787
22. Kokkinos, I., Yuille, A.L.: HOP: Hierarchical object parsing. In: CVPR. (2009)
23. Ling, H., Jacobs, D.: Shape classification using the inner-distance. IEEE TPAMI **29**(2) (2007) 286–299
24. Torsello, A., Robles-Kelly, A., Hancock, E.R.: Discovering shape classes using tree edit-distance and pairwise clustering. IJCV **72**(3) (2007) 259–285
25. Trinh, N.H., Kimia, B.B.: Skeleton search: Category-specific object recognition and segmentation using a skeletal shape model. IJCV **94**(2) (2011) 215–240
26. Bai, X., Wang, X., Liu, W., Latecki, L.J., Tu, Z.: Active skeleton for non-rigid object detection. In: ICCV. (2009)
27. Shotton, J., Blake, A., Cipolla, R.: Multiscale categorical object recognition using contour fragments. PAMI **30**(7) (July 2008) 1270–81
28. Ferrari, V., Tuytelaars, T., Gool, L.V.: Object detection by contour segment networks. In: ECCV. (2006) 14–28
29. Perona, P., Malik, J.: Detecting and localizing edges composed of steps, peaks and roofs. In: ICCV. (1991) 52–57
30. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE TPAMI **33** (2011) 898–916

31. Felzenszwalb, P., McAllester, D.: A min-cover approach for finding salient curves. In: IEEE Workshop on Perceptual Organization (POCV). (2006)
32. Russell, B., Freeman, W., Efros, A., Sivic, J., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: CVPR. (2006)
33. Todorovic, S., Ahuja, N.: Unsupervised category modeling, recognition, and segmentation in images. IEEE TPAMI **30**(12) (2008) 1–17
34. Kim, G., Faloutsos, C., Hebert, M.: Unsupervised modeling of object categories using link analysis techniques. In: CVPR. (2008)
35. Lee, Y.J., Grauman, K.: Shape discovery from unlabeled image collections. In: CVPR. (2009)
36. Felzenszwalb, P., McAllester, D.: A min-cover approach for finding salient curves. In: CVPR POCV. (2006)
37. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. PAMI **24**(4) (2002) 509–522
38. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60**(2) (2004) 91–110
39. Chib, S., Greenberg, E.: Understanding the metropolis-hastings algorithm. The American Statistician **49**(4) (1995) 327–335
40. Lin, L., Zeng, K., Liu, X., Zhu, S.C.: Layered graph matching by composite cluster sampling with collaborative and competitive interactions. In: CVPR. (June 2009)
41. Lee, Y.J., Grauman, K.: Foreground focus: Unsupervised learning from partially matching images. In: BMVC. (2008)
42. Fei-Fei L., F.R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: CVPR. (2004)
43. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. Technical Report AIM-2005-025, MIT (2005)
44. Borenstein, E., Ullman, S.: Class-specific, top-down segmentation. In: ECCV. Volume 2. (2002) 109–124

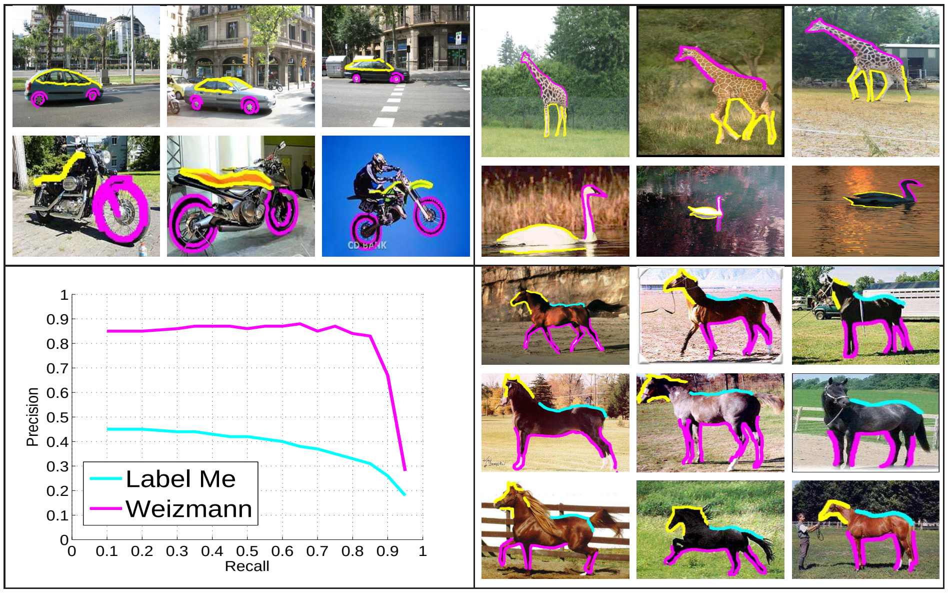


Fig. 6. Unsupervised detection and segmentation of objects in example images from LabelMe (top left), ETHZ (top right), and Weizmann Horses (bottom right). For LabelMe and ETHZ, each row shows images that are grouped within a unique image cluster by CASW in the first stage. Contours that are clustered by CASW in the second stage are highlighted with distinct colors indicating cluster membership. CASW accurately discovers foreground objects, and delineates their characteristic parts. E.g., for LabeMe *Cars sideview* CASW discovers two contour clusters (yellow and magenta), corresponding to the two car parts wheels and roof. (bottom left) ROC curves for LabelMe and Weizmann Horses, obtained by varying the minimum allowed DTW similarity between pairs of matching contours which are input to CASW. (best viewed in color)