In Proceedings of the European Conference on Computer Vision (ECCV), 2012.

Shape Sharing for Object Segmentation

Jaechul Kim and Kristen Grauman

Department of Computer Science, The University of Texas at Austin

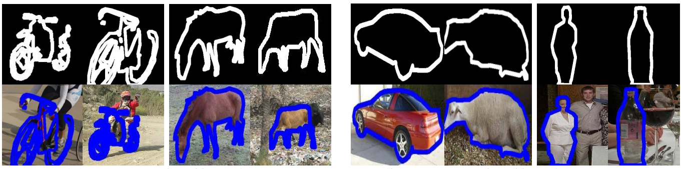
Abstract. We introduce a category-independent shape prior for object segmentation. Existing shape priors assume class-specific knowledge, and thus are restricted to cases where the object class is known in advance. The main insight of our approach is that shapes are often shared between objects of *different* categories. To exploit this "shape sharing" phenomenon, we develop a non-parametric prior that transfers object shapes from an exemplar database to a test image based on local shape matching. The transferred shape priors are then enforced in a graph-cut formulation to produce a pool of object segment hypotheses. Unlike previous multiple segmentation methods, our approach benefits from global shape cues; unlike previous top-down methods, it assumes no class-specific training and thus enhances segmentation even for unfamiliar categories. On the challenging PASCAL 2010 and Berkeley Segmentation datasets, we show it outperforms the state-of-the-art in bottom-up or category-independent segmentation.

1 Introduction

Bottom-up image segmentation methods group low-level cues from color, texture, and contours to estimate the boundaries in an image. Despite significant strides in recent years, it is widely acknowledged that a bottom-up process alone cannot reliably recover object-level segments. Pitfalls include the fact that a single object is often comprised of heterogeneous textures and colors, objects with similar appearance can appear adjacent to one another, and occlusions disrupt local continuity cues—all of which lead to over- or under-segmented results. This can be a fatal flaw for downstream recognition processes.

As a result, researchers have explored two main strategies to move beyond low-level cues. The first strategy expands the output to produce *multiple segmentation hypotheses*, typically by using hierarchical grouping, varying hyperparameters, or merging adjacent regions (e.g., [1–5]). Enlarging the set of segments increases the chance of "hitting" a true object; however, large pools of candidate regions are costly to compute and maintain, and, more importantly, existing methods lack a model of global shapes.[1] The second strategy introduces top-down *category-specific priors*, unifying bottom-up evidence with a preference to match a particular object's layout, shape, or appearance (e.g., [6–9]). Such methods elegantly integrate segmentation and recognition, yet they rely heavily on

[1] Throughout, we use *shape* to refer to the outer contours or boundaries of objects.



(a) Semantically *close* (b) Semantically *disparate*

Fig. 1. Intuition for shape sharing. While one may expect shape sharing between objects of semantically close categories (a), we observe that similar shapes exist even among semantically disparate objects (b). This suggests transferring "object-level" shapes between categories, to enable *category-independent* shape priors.

a known (pre-trained) class model. Further, category-specific shape priors often make strong assumptions about the viewpoint of the object to be segmented.

At the surface, the goals of these two existing strategies seem to be in conflict: the former maintains category-independence, while the latter enforces top-down shape knowledge. Our idea is to reconcile these competing goals by developing a category-independent shape prior for segmentation. The main insight of our approach is that similar shapes exist across objects—and this "shape sharing" occurs even across disparate categories. See Figure 1. Thus, rather than learn a narrow prior good only on the known class of interest, we can transfer object shapes between classes, thereby leveraging top-down shape cues to segment objects *regardless of their category.*

To this end, we propose a non-parametric, data-driven prior based on partial shape matching. Given a novel unsegmented image, we first extract local shapes using a boundary-preserving local region detector, and then identify any strong matches it has with shapes in a database of segmented exemplars. Based on the scale and position of each local shape match, we project the associated exemplar shapes into the test image. This effectively maps local support into global shape hypotheses without assuming any category-specific knowledge, since the database need *not* contain exemplars of the same object class(es) as our test image. Each set of highly overlapping shape projections yields a shape prior for the novel image, suggesting regions that would not be considered if judging color/texture alone. Finally, we generate multiple output segment hypotheses by performing a series of figure-ground segmentations using graph-cuts, enforcing each of the shape priors in turn. Figure 2 overviews the approach.

Results on the PASCAL 2010 and Berkeley Segmentation datasets show that our approach outperforms not only bottom-up segmentation [10], but also state-of-the-art category-independent region generation methods that lack shape priors [3, 4]. Furthermore, we demonstrate it is even competitive with an analogous category-specific shape prior, lending clear support for shape sharing among different categories. As such, unlike existing top-down segmentation methods, our approach can enhance the segmentation of objects it has never seen previously.

2 Related Work

In segmentation, shape is typically used as a category-specific cue, whereby known object models are integrated with bottom-up grouping cues [6, 7, 11, 12]. In contrast, our approach leverages shape in a category-independent manner, and thus does not require prior knowledge about the object(s) present.

The notion of sharing visual properties has been pursued in various forms in computer vision. In object detection, jointly training multi-class detectors allows the reuse of common discriminative features [13, 14]. In image parsing, hierarchical representations can exploit local parts shared between objects [15, 16]. In object recognition, model parameters learned on one set of categories can be transferred to more efficiently learn new related objects [17, 18]. All such prior methods focus on sharing features to reduce redundancy and increase computational efficiency, and they employ category-labeled data to explicitly train shared parts/features. In contrast, we propose a novel form of sharing to estimate shape priors for image segmentation, and our data-driven method uses no class labels.

For shapes in particular, knowledge about shared properties is typically expressed in parametric forms, e.g., Gestalt cues like symmetry [19], or hand-crafted geometric primitives [20]. A recent method for figure-ground contour classification discovers prototypical local geometric features, yet it depends on bottom-up cues alone when grouping the labels predicted by each local prototype [21]. In contrast, we consider *object-level* sharing, whose top-down nature allows our method to globally group parts of diverse appearance. Unlike any of the above, we propose an exemplar-based, non-parametric approach to sharing, which offers flexibility to the rich variations of object shapes and poses.

Exemplar-based methods have long been explored in vision and graphics. Some recent work tackles segmentation in a data-driven manner [22, 23], using image-level matching to gather exemplars with similar scene layouts, and then combining them with graph-cuts to preserve spatial coherence. Their image-level matching is too coarse to capture individual objects' shapes and is sensitive to scale and position changes; they are therefore most applicable to the images with a single object or consistent scene layout. A contemporary approach uses window-level matching for more robust exemplar retrieval under image variations [24]. However, the window matching can be distracted by background clutter when an object's shape does not fit in the window and so the window is dominated by background pixels. In contrast to all these methods, our method retrieves exemplars according to local shape matches, which allows it to delineate multiple objects in spite of spatial layout variations and/or background clutter.

Also relevant to our work are recent methods that generate category-independent object segmentation hypotheses [3, 4]. Like our method, they also assume access to a database of segmented images, generate multiple object hypotheses using similar multi-parametric graph-cuts, and offer improvements over purely bottom-up image segmentation. However, the previous techniques rely on local bottom-up cues (color, texture, contour strengths). Their lack of shape priors hurts performance—particularly for cases where color consistency is insufficient to form a good segment, as we show in the results.

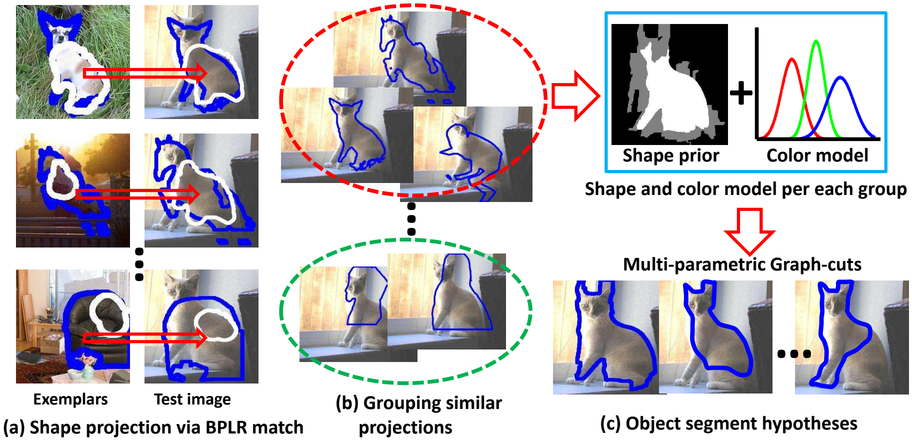

Exemplars
Test image
(a) Shape projection via BPLR match
(b) Grouping similar projections
Shape prior
Color model
Shape and color model per each group
Multi-parametric Graph-cuts
(c) Object segment hypotheses

Fig. 2. Overview of our method. (a) Exemplars (first column) that share shape with the test image (second column) are projected in, no matter their category. We identify those shared shapes (marked in blue) via local BPLR matches (marked in white). (b) Multiple exemplars that highly overlap are aggregated, to form a shape prior and color model per each aggregated group. (c) The priors from each group are used to compute a series of graph-cut segmentation hypotheses.

3 Approach

The input to our method is an unsegmented image containing unknown object categories, and the output is a set of object segment hypotheses (which may overlap). The method is successful to the extent that the hypotheses contain regions that highly overlap with true object boundaries.

Our approach consists of three main steps: (1) estimating global object shape in a test image by projecting exemplars via local shape matches (Sec. 3.1), (2) aggregating sets of similarly aligned projected shapes to form a series of hypothesized shape priors (Sec. 3.2), and (3) enforcing the priors within graph-cuts to generate object segment hypotheses (Sec. 3.3).

3.1 Projecting Global Shapes from Local Matches

Suppose we have a database of manually segmented exemplars of a variety of objects. For each exemplar object, we extract a set of distinctive local region features. We use the Boundary-Preserving Local Region (BPLR) method to detect the local shape regions [25]; it is a publicly available dense local feature detector whose boundary-preserving property is well-suited to shape matching. To describe the shape of each detected region, we extract a pHOG descriptor computed on a gPb [10] contour map, which captures both boundary shape and coarse inner texture.

Given a test image, the goal is to identify with which exemplars it shares shape. We first extract BPLRs throughout the test image, generating a dense set

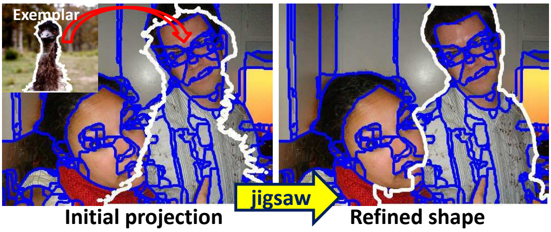


Fig. 3. Jigsaw puzzling the superpixels underlying the exemplar's projection.

of local shape regions (∼1,000 per image). Then, we match each BPLR in the test image to the exemplar database by finding its $k = 5$ nearest neighbor descriptors among all of the exemplars' BPLRs. For each such *local* shape match, we project the associated exemplar's *global* outer boundary shape into the test image based on the similarity transform (scale and translation) computed between the two matched features (see Figure 2(a)). The density of the BPLR detector establishes thousands of such initial global shape projections per test image.

Due to shape deformations and uncertainty in the local match, however, the projected shapes need not be entirely aligned with the test image's contours. Therefore, we next want to snap the projected shape to align with bottom-up evidence of boundaries. To this end, we refine the initial projection boundary to span the "jigsaw" of underlying superpixels that overlap the global shape by more than half their total area. In this way, the exemplar shape is adapted to fit the observed contours. Figure 3 shows an example jigsaw puzzling.

Finally, we eliminate unreliable projections whose shape changes substantially after the refinement process. Specifically, we rank them by the pixel-level overlap between the original exemplar's projection and the jigsaw refined version, and select the top-ranked projections. Essentially this weeds out unreliable projections that lack bottom-up support in the test image. In our implementation, we keep the top 600 projections from the initial ∼5,000 candidates.

The key impact of this step is to generate *globally* shared boundary hypotheses from *locally* matched shapes. Broadly speaking, the use of local matches to generate global contour estimates has long been considered in vision, e.g., in early model-based recognition [26] or voting-based implicit shape models [27]. However, the key novelty of our design is its category-independence. Whereas existing approaches seek matches of the same instance or category, our predictions are made *across* categories. In fact, it is irrelevant to our method whether or not the exemplar shapes have class labels; their value is solely in providing a non-parametric prior for what kinds of shapes objects take on.

3.2 Aggregating Partially Shared Shapes

At this point, we could simply treat each of the global shape projections computed above as an individual shape prior; in fact, we find that alone they provide a reasonable prior (see Table 1 in results). However, doing so would not account for the fact that objects in the test image often share shapes with various exemplars—some of them only partially. Therefore, we next aim to group together those shape projections that agree on their spatial extents in the test image. The idea is for each projection to contribute a portion of its contour to an aggregate shape prior (e.g., see the matched exemplars in Figure 2(a), each of which partially shares shape with the cat in the test image). In addition, the aggregation removes redundancy among the highly overlapping projections, which in effect reduces the number of shape priors for the subsequent graph-cuts computation.

To determine which projections to aggregate, we use a simple but effective metric: any projections whose pixel overlap exceeds 50% are grouped. (In practice, this typically yields 250-300 groups given $\sim$600 individual shape projections.) Each such group is used to construct one shape prior consisting of two parts: one that prefers including those pixels in the test shape that are shared by the contributing exemplar projections, and one that extracts a color model using their predicted shape. See Figure 2(b) and (c). Both parts enforce the shape prior in a graph-cut figure-ground segmentation, as we explain next.

3.3 Graph-Cut Segmentation with the Shape Prior

The final step is to enforce the non-parametric priors when computing the output region hypotheses. We define an energy function that measures the quality of a given figure-ground segmentation according to its agreement with the shape prior. We optimize this function independently for each group (prior) defined above, yielding one set of region hypotheses per group.

Treating each pixel p_i in the image as a node, graph-cut optimizes their labels $y_i \in \{0 \text{ (bg)}, 1 \text{ (fg)}\}$ by minimizing an energy function of the form:

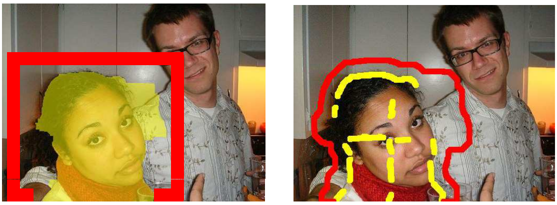
$$E(y) = \sum_{p_i \in \mathcal{P}} D_i(y_i) + \sum_{i,j \in \mathcal{N}} V_{i,j}(y_i, y_j), \tag{1}$$

where $\mathcal{P}$ denotes all pixels, $\mathcal{N}$ denotes pairs of adjacent pixels, $V_{i,j}$ is a smoothness function, and D_i is a data term. Note that this follows the basic graph-cut segmentation formulation; what we focus on is how to encode a non-parametric shape prior into the data term.

Data term: Typically, the data term D_i is a function of the likelihood of labeling pixel p_i as foreground or background. In our formulation, it consists of two parts: a shape-prior likelihood S_i and a color likelihood C_i: $D_i(y_i) = S_i(y_i) + C_i(y_i)$.

The shape-prior term S_i defines the likely spatial extent of the foreground and background. Given one group from Sec. 3.2, we first compute the intersection $\mathcal{I}$ and union $\mathcal{U}$ of its component shape projection regions. Then we define the cost of labeling a pixel as foreground to be:

$$S_i(y_i = 1) = \begin{cases} 0.5 - \gamma & \text{if } p_i \in \mathcal{I} \\ 0.5 + \gamma & \text{if } p_i \notin \mathcal{U} \\ 0.5 & \text{if } p_i \notin \mathcal{I} \text{ and } p_i \in \mathcal{U}, \end{cases} \tag{2}$$



(a) Region-based seeds (b) Contour-based seeds

Fig. 4. The two methods for constructing H_f and H_b histograms. Yellow: fg seeds, Red: bg seeds. Best viewed in color.

where $p_i \in \mathcal{I}$ and $p_i \in \mathcal{U}$ denote a pixel inside the intersection and union of the projections, respectively, and γ is a positive constant value used to adjust the impact of the shape prior (and will be defined below). The cost of assigning the background label is simply the inverse: $S_i(y_i = 0) = 1 - S_i(y_i = 1)$. Intuitively, this likelihood prefers a pixel inside the intersection region to be labeled as foreground, since all of the projections in the group agree that the pixel belongs in the shape. In contrast, it prefers a background label for pixels outside the union region, since none of the projections predict the pixel to belong to the shape (i.e., no sharing). Pixels in the union but outside of the intersection are treated as neutral, with no bias towards either foreground or background, as reflected by the third line in Eqn. 2. The white and gray pixels in Figure 2(b) depict these foreground biased and "don't care" regions of the shape prior, respectively.

The color likelihood term C_i also relies on the shape projections, but in a different way. Whereas S_i biases pixel memberships based on the span of the shared shapes, C_i uses the shared shape to estimate a color distribution for the hypothesized object. Let H_f and H_b denote normalized color histograms sampled from the shared shape region for the foreground and background, respectively. We define the color likelihood cost as:

$$C_i(y_i) = \frac{1}{1 + \exp(\beta W_i(y_i))}, \tag{3}$$

where $W_i(p_i)$ is a function of the color affinity between pixel p_i and the histograms, and β is a normalizing constant that is automatically determined as the inverse of the mean value of W_i over all pixels. Let $c(p_i)$ denote the histogram bin index of the RGB color value at pixel p_i. The color affinity rewards assigning the background label to pixels more likely to be generated by the background color distribution:

$$W_i(y_i = 0) = H_b(c(p_i)) - H_f(c(p_i)), \tag{4}$$

and vice versa: $W_i(y_i = 1) = -W_i(y_i = 0)$. The sigmoid in Eqn. 3 serves to scale the color likelihoods between 0 and 1, making them compatible with the shape-prior values S_i.

We devise two complementary ways to sample pixels from the shared shape in order to form H_f and H_b: one that uses *region-based seed pixels*, and one that uses *contour-based seed pixels*. For region-based seed pixels, H_f is computed using all pixels inside the intersection $\mathcal{I}$ of the shape projections, and H_b is computed using pixels falling within a thick rectangular border surrounding the intersection region. See Figure 4(a). For contour-based seed pixels, we instead form H_f using pixels along the boundary of $\mathcal{I}$ and along its primary medial axes within the shape, and we compute H_b using pixels along the boundary of a dilated version of the same intersection region. We ignore any boundary pixels having weak gPb [10] values. See Figure 4(b).

The two seeding methods work in complementary ways. Region-based seeding provides dense coverage of pixels, and thus reflects the full color distribution of the shape prior's region. However, when the shape prior is flawed—for example, if it leaks into the background, as shown in Fig. 4(a)—then its estimate can be distorted. On the other hand, contour-based seeding respects the object shapes, and is motivated by how users tend to manually give seeds for interactive segmentation [28]. However, being sparser, it may lack sufficient statistics to estimate the color distribution. We use each of these seeding strategies separately when generating the pool of segmentations (see below).

Smoothness term: Our smoothness function $V_{i,j}$ follows the conventional form, e.g.,[28]: the cost of assigning different labels to neighboring pixels is inversely proportional to the strength of the contour at that position.

Multi-parametric graph-cuts: Having defined the complete energy function $E(y)$, we can now compute the optimal binary labeling using graph-cuts. For each group of projections resulting from Sec. 3.2, we solve *multiple* instances of the problem by varying the weighting constants and color histogram seeding strategies. This yields multiple segment hypotheses for a given prior, and is in the common spirit of the sequence of parametric min-cuts performed in [3, 4] (though, our focus is to incorporate the shape prior).

Specifically, we vary (1) the value of γ in Eqn. 2, which adjusts the influence of the shape prior relative to the color likelihood, (2) whether region-based or contour-based seeding is used, which adjusts the definition of C_i in Eqn. 3, and (3) the value of a foreground bias constant λ in the data term. For the latter, we modify the data term D_i as follows:

$$D_i(y_i, \lambda) = \begin{cases} D_i(y_i) + \lambda & \text{if } y_i = 1 \\ D_i(y_i) - \lambda & \text{if } y_i = 0. \end{cases} \tag{5}$$

Positive values of λ decrease the foreground bias, while negative ones raise it.

Thus, the total number of hypotheses for the given group is (#γ values) $\times 2 \times$ (#λ values); we use 2 and 8 values of γ and λ in our experiments, respectively. Note that increasing the pool of segments naturally will increase recall of true object shapes, but at the penalty of greater complexity.

4 Results

The main goals of the experiments are (1) to demonstrate that shape sharing improves the quality of the segmentation (Sec. 4.1), (2) to analyze under what

Approach	Covering (%)	Num segments
Exemplar-based merge (Ours)	**77.0**	607
Neighbor merge [2]	72.2	5005
Bottom-up segmentation [10]	62.8	1242

Table 1. Our shape-based projection and merging approach outperforms an existing merging strategy while requiring an order of magnitude fewer segments (second row). It also substantially improves the state-of-the-art bottom-up segmentation (third row).

conditions shapes are useful for segmentation (Sec. 4.2), and (3) to validate the impact of our category-independent shape priors compared to traditional category-dependent ones (Sec. 4.3).

Datasets and implementation details: To build the exemplar database, we use the PASCAL 2010 Segmentation training data, which has pixel-level annotations for 2,075 objects from 20 classes. We extract 1,000-2,000 BPLRs from each exemplar, and represent them with pHOG+gPb descriptors. To efficiently identify nearest neighbor matches, we use FLANN [29]. For superpixels, we use the output of gPb-owt-ucm [10]. We sample values for γ in Eqn. 2 and λ in Eqn. 5 uniformly, following [3, 4].

It takes about 4 minutes to generate hypotheses in our unoptimized Matlab code (5 sec to match BPLRs + 50 sec to project shapes + 3 mins for graph-cuts). The best competing methods [3, 4] also use multi-parametric graph-cuts; so, the additional time required by our method is fairly small and could be reduced further by parallelizing the shape projection step.

We test on two datasets: the PASCAL 2010 validation set and the Berkeley BSD300 dataset. For BSD, we use the ground truth region annotations given by [4]. Note that for both test sets, we use the same PASCAL exemplars.

Evaluation metrics: To evaluate segmentation quality, we use the *covering metric*, following [10, 3], which is the average best overlapping score between ground-truth and generated segments, weighted by object size. Note that due to the use of "best overlap" in the covering metric, a method that achieves higher covering for fewer segments has better focused its results on true object regions. We also report *recall as a function of overlap*, following [4], to quantify the percentage of objects recalled at a given covering score.

4.1 Segmentation Quality

First we investigate how well shape sharing improves segmentation accuracy, by comparing our results to those of several state-of-the-art techniques [10, 2–4].

Considering the shape priors alone: First we evaluate the quality of our exemplar-based shape priors (i.e., the first stage of our method defined in Sec. 3.1). We compare against two existing methods on the PASCAL data: (1) a merging method that combines pairs and triples of neighboring superpixels, without considering layout or shape [2], and (2) the state-of-the-art bottom-up hierarchal segmentation algorithm [10]. Both are important baselines, since [2]

Approach	Covering (%)	Num segments
Shape Sharing (Ours)	**84.3**	1448
CPMC [3]	81.6	1759
Object proposals [4]	81.7	1540
gPb-owt-ucm [10]	62.8	1242

Table 2. Accuracy on the PASCAL2010 dataset.

Approach	Covering (%)	Num segments
Shape Sharing (Ours)	**75.6**	1449
CPMC [3]	74.1	1677
Object proposals [4]	72.3	1275
gPb-owt-ucm [10]	61.6	1483

Table 3. Accuracy on the BSD300 dataset.

also entails merging superpixels but lacks top-down shape cues, while [10] provides the original regions to both merging methods.

Table 1 shows the results. Our method clearly outperforms the previous methods, while also maintaining a much smaller number of segments. This confirms the ability of shape sharing to predict the objects' spatial extent.

Final segmentation with graph-cuts: Next we compare our full approach to existing segmentation methods, including the state-of-the-art category-independent object segmentation generators of [3] and [4]. We use the code kindly provided by the authors.[2] To focus on raw segmentation quality, we do not consider post-processing with a learned region-ranking function (as in [3], [4]), which could equally benefit all methods, in terms of the number of segments.

Tables 2 and 3 show the results on PASCAL and BSD, respectively. Our approach outperforms the existing methods. It is also more accurate for 18 of the 20 PASCAL classes, with per-class gains up to 9 points (see supp. file[3]). Since all three previous methods rely on only color and/or local appearance and layout cues, this result validates the impact of global shape priors for segmentation.

The strength of our method on BSD—for which we use PASCAL images as exemplars—is strong evidence that shape sharing is generalized among various objects of unrelated categories. Even the PASCAL test results illustrate category-independence, since the exemplars matched to test images can and often do come from different categories. When examining the sharing strength between all pairs of PASCAL object classes, we find that shape sharing often occurs among semantically close categories (e.g., among animals or vehicles) as well as semantically disparate classes (e.g., bottle and person); see Figure 5. In Sec. 4.3 below we further explicitly isolate the impact of category-independence.

[2] In order to isolate the impact of a color-based graph-cut likelihood, for [3], we select an option in the author's code to forgo graph-cut outputs with uniform foreground bias, which do not rely on image cues.

[3] http://vision.cs.utexas.edu/projects/shapesharing/supp.pdf

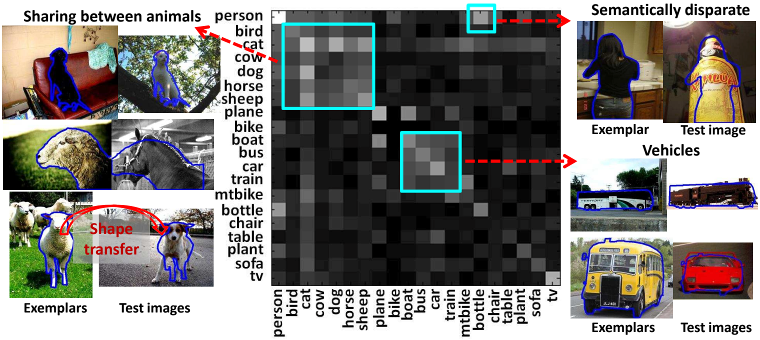


Fig. 5. Shape sharing matrix for the 20 classes in PASCAL. We compute the strength of sharing by counting how many times shape exemplars from one class are used to generate the best segmentation hypotheses for another class. Brighter=higher.

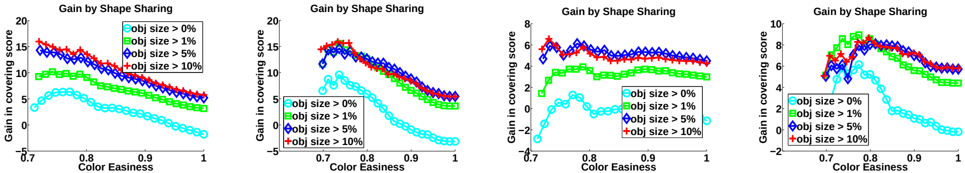
4.2 Impact of Shapes on Segmentation

Thus far, we have presented the results in terms of the average score over all test objects and classes. To go beyond this summary of overall performance, we next dig deeper to see where and why shape sharing is most effective. That is, under what conditions will our shape prior most benefit segmentation? We expect shape to serve a complementary role to color, and to be most useful for objects that consist of multiple parts of diverse colors, and for objects that are similarly colored to nearby objects and the background.

To validate this hypothesis, we introduce a measure of *color easiness*, such that we can rank all test images by their expected amenability to color-based segmentation. We define color easiness by building fg and bg color histograms using pixels from inside and outside the ground truth object boundaries, respectively, and then count how many pixels in the object's bounding box would be correctly labeled if using only their distance to the two histograms. The more correctly labeled pixels, the higher the color easiness for that test image.

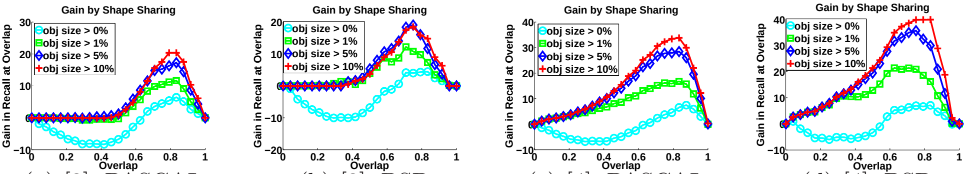
Figure 6 plots Shape Sharing's accuracy gain over the baselines, as a function of color easiness (x-axis) and object size (multiple curves per plot). We see clearly that the most impressive gains—up to about 15 points in raw covering score—indeed occur when color easiness is lowest, for both datasets. The trend with color easiness is especially pronounced in the comparison to [3] (see (a) and (b)), which makes sense because its cues are strictly color-based. In contrast, compared to [4] the trend is a bit flatter, since that method uses not only color but also a local layout cue (see (c) and (d)). Still, our gains are substantial over both methods.

Figure 6 also reveals that Shape Sharing most benefits the segmentation of larger objects. In fact, the average gain in covering score increases from 2.7 points (Table 2, for all objects) to 4.8 points for non-trivial objects in size that are larger than 2% of image size (∼40 by 40 pixels). We attribute this to a couple of factors. First, shapes become more evident as object size increases, since there is



(a) [3], PASCAL (b) [3], BSD (c) [4], PASCAL (d) [4], BSD

Fig. 6. Impact of Shape Sharing as a function of "color easiness". When color alone is most confusing (low easiness), Shape Sharing shows the greatest accuracy gains.



(a) [3], PASCAL (b) [3], BSD (c) [4], PASCAL (d) [4], BSD

Fig. 7. Shape Sharing's gain in recall as a function of overlap.

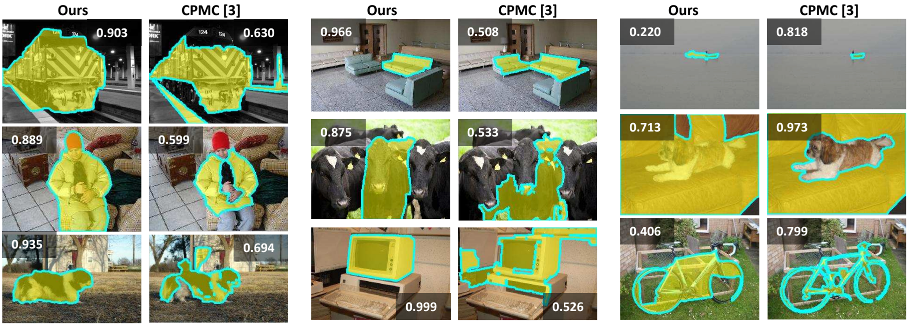
sufficient resolution along the boundary. Second, since larger objects tend to have various parts with diverse colors (e.g., a close-up of a person wearing differently colored pants and shirt), shape becomes more critical to combine the disparate parts. On the other hand, Shape Sharing has little impact (and can even hurt accuracy) for the smallest objects that occupy less than 1% of the image. This is because local shape matches are missed on the tiny objects, or the scale change computed from the local match becomes unreliable.

Figure 7 plots Shape Sharing's gain in recall as a function of overlap score, where recall records what percentage of objects have a best overlap score over the given threshold. Ours outperforms the baselines. In particular, our method provides the greatest gains in what is arguably a critical operating range for segmentation: overlaps from about 0.6-0.9. Why is this a critical range? For overlaps beyond 0.9, many segmentations are so easy as to make the perceived "winner" a toss-up. On the other hand, for low overlaps less than 0.5, segmentations are all poor, similarly making it hard to perceive the difference. However, in the range of about 0.6 to 0.9, segmentation quality is reasonable while images contain substantial challenges for segmentation, making the qualitative differences among methods much more evident. See supp. file for an illustration.

Figure 8 show example results from our method and the best competing method, CPMC [3], illustrating when the shape prior is most beneficial.

4.3 Category-Independent vs. Category-Specific

Finally, we directly study the extent to which our method's success is based on its category-independence. We compare Shape Sharing to two baselines. The first is a *category-specific* approach that operates just as our method, *except* that only exemplars of the same class as the test instance may be used (which, of



(a) Objects with diverse colors (b) Objects similarly colored as surroundings (c) Failure cases

Fig. 8. Results from Shape Sharing (left col per group) and CPMC [3] (right col per group). (a) Shapes pull together diversely-colored parts of an object. (b) Shapes help delineate an object from surroundings of similar colors: e.g., nearby objects from the same class (rows 1, 2), or confusing backgrounds (3rd row). (c) Shapes do not help segment tiny objects (1st row), nor objects lacking shape, e.g., the truncated sofa (2nd row), or thin structured objects like the bicycle (3rd row).

Approach	Covering (%)
Category-specific	**84.7**
Category-independent (Default)	84.3
Strictly category-independent	83.9
CPMC [3]	81.6
Object proposals [4]	81.7

Table 4. Comparison of category-independent and category-specific variants of our approach on PASCAL data.

course, uses information that would not be available in most realistic scenarios). The second is a *strictly category-independent* variant, where we require that the exemplar matched to a test image *must* be from another class; this too is not enforceable in realistic settings, but it verifies our gains are *not* due to having segmented exemplars of the same object class available.

Table 4 shows the results, with the previous baseline numbers repeated for reference in the bottom two rows. As expected, the category-specific variant performs best, and strictly-independent performs worst. However, the accuracy of all three is quite close. In addition, even our strictly independent variant outperforms the previous baselines that lack shape priors. This result demonstrates that shapes are truly shared among different categories, and one can use the proposed shape priors in a category-independent manner; we do not need hand-crafted exemplars for the object(s) in the test image for segmentation.

5 Conclusions

We introduced a category-independent shape prior for segmentation that exploits shape sharing between objects of different categories. Through extensive experiments, we showed that (1) shape sharing improves the quality of bottom-up segmentation, while requiring no prior knowledge of the object, and (2) our category-independent prior performs as well as a parallel category-specific one, demonstrating that shapes are truly shared across categories. As such, unlike previous top-down segmentation methods, our approach can enhance the segmentation of previously unseen objects.

Acknowledgements: This research is supported in part by ONR grant N000141210068.

References

[1] Hoiem, D., Efros, A., Hebert, M.: Geometric context from a single image. In: ICCV. (2005)

[2] Malisiewicz, T., Efros, A.: Improving Spatial Support for Objects via Multiple Segmentations. In: BMVC. (2007)

[3] Carreira, J., Sminchisescu, C.: Constrained Parametric Min-Cuts for Automatic Object Segmentation. In: CVPR. (2010)

[4] Endres, I., Hoiem, D.: Category Independent Object Proposals. In: ECCV. (2010)

[5] van de Sande, E., Uijlingsy, J., Gevers, T., Smeulders, A.: Segmentation as Selective Search for Object Recognition. In: ICCV. (2011)

[6] Borenstein, E., Ullman, S.: Class-Specific, Top-Down Segmentation. In: ECCV. (2002)

[7] Levin, A., Weiss, Y.: Learning to Combine Bottom-Up and Top-Down Segmentation. In: ECCV. (2006)

[8] Mairal, J., Leordeanu, M., Bach, F., Hebert, M., Ponce, J.: Discriminative Sparse Image Models for Class-Specific Edge Detection and Image Interpretation. In: ECCV. (2008)

[9] Chan, T., Zhu, W.: Level Set Based Shape Prior Segmentation. In: CVPR. (2005)

[10] Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour Detection and Hierarchical Image Segmentation. PAMI **33** (2011)

[11] Vu, N., Manjunath, B.: Shape Prior Segmentation of Multiple Objects with Graph Cuts. In: CVPR. (2008)

[12] Brox, T., Bourdev, L., Maji, S., Malik, J.: Object Segmentation by Alignment of Poselet Activations to Image Contours. In: CVPR. (2011)

[13] Torralba, A., Murphy, K., Freeman, W.: Sharing Visual Features for Multiclass and Multiview Object Detection. PAMI **29** (2007)

[14] Opelt, A., Pinz, A., Zisserman, A.: Incremental Learning of Object Detectors Using a Visual Shape Alphabet. In: CVPR. (2006)

[15] Zhu, L., Chen, Y., Yuille, A., Freeman, W.: Latent Hierarchical Structure Learning for Object Detection. In: CVPR. (2010)

[16] Fidler, S., Boben, M., Leonardis, A.: Similarity-Based Cross-Layered Hierarchical Representation for Object Categorization. In: CVPR. (2008)

[17] Fei-Fei, L., Fergus, R., Perona, P.: A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories. In: ICCV. (2003)

[18] Stark, M., Goesele, M., Schiele, B.: A Shape-based Object Class Model for Knowledge Transfer. In: ICCV. (2009)

[19] Levinshtein, A., Sminchisescu, C., Dickinson, S.: Multiscale Symmetric Part Detection and Grouping. In: ICCV. (2009)

[20] Biederman, I.: Recognition-by-Components: A Theory of Human Image Understanding. Psychological Review **44** (1987) 115–147

[21] Ren, X., Fowlkes, C., Malik, J.: Figure/Ground Assignment in Natural Images. In: ECCV. (2006)

[22] Russell, B., Efros, A., Sivic, J., Freeman, W., Zisserman, A.: Segmenting Scenes by Matching Image Composites. In: NIPS. (2009)

[23] Rosenfeld, A., Weinshall, D.: Extracting Foreground Masks towards Object Recognition. In: ICCV. (2011)

[24] Kuettel, D., Ferrari, V.: Figure-Ground Segmentation by Transferring Window Masks. In: CVPR. (2012)

[25] Kim, J., Grauman, K.: Boundary Preserving Dense Local Regions. In: CVPR. (2011)

[26] Rothwell, C., Zisserman, A., Forsyth, D., Mundy, J.: Canonical Frames for Planar Object Recognition. In: ECCV. (1992)

[27] Leibe, B., Leonardis, A., Schiele, B.: Combined Object Categorization and Segmentation with an Implicit Shape Model. In: Workshop on Statistical Learning in Computer Vision. (2004)

[28] Rother, C., Komogorov, V., Blake, A.: Grabcut: Interactive Foreground Extraction Using Iterated Graph Cuts. In: SIGGRAPH. (2004)

[29] Muja, M., Lowe, D.: Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In: VISAPP. (2009)