Object-Graphs for Context-Aware Visual Category Discovery

Yong Jae Lee, *Student Member, IEEE*, and Kristen Grauman, *Member, IEEE*

Abstract—How can knowing about some categories help us to discover new ones in unlabeled images? Unsupervised visual category discovery is useful to mine for recurring objects without human supervision, but existing methods assume no prior information and thus tend to perform poorly for cluttered scenes with multiple objects. We propose to leverage knowledge about previously learned categories to enable more accurate discovery, and address challenges in estimating their familiarity in unsegmented, unlabeled images. We introduce two variants of a novel object-graph descriptor to encode the 2D and 3D spatial layout of object-level co-occurrence patterns relative to an unfamiliar region and show that by using them to model the interaction between an image's known and unknown objects, we can better detect new visual categories. Rather than mine for all categories from scratch, our method identifies new objects while drawing on useful cues from familiar ones. We evaluate our approach on several benchmark data sets and demonstrate clear improvements in discovery over conventional purely appearance-based baselines.

Index Terms—Object recognition, context, category discovery, unsupervised learning.

1 Introduction

The goal of unsupervised visual category learning is to take a completely unlabeled collection of images and discover those appearance patterns that repeatedly occur in many examples. Often these patterns will correspond to object categories or parts, and the resulting clusters or visual "themes" are useful to summarize the images' content or to build new models for object recognition using minimal manual supervision [1], [2], [3], [4], [5]. The appeal of unsupervised methods is threefold: First, they help reveal structure in a very large image collection; second, they can greatly reduce the amount of effort that currently goes into annotating or tagging images; and third, they mitigate the biases that inadvertently occur when manually constructing data sets for recognition. The potential reward for attaining systems that require little or no supervision is enormous, given the vast (and ever increasing) unstructured image and video content currently available—for example, in scientific databases, news photo archives, or on the web.

Existing unsupervised techniques essentially mine for frequently recurring appearance patterns, typically employing a clustering algorithm to group local features across images according to their texture, color, shape, etc. Unfortunately, learning multiple visual categories simultaneously from unlabeled images remains understandably difficult, especially in the presence of substantial clutter and scenes with multiple objects. While appearance is a fundamental cue for recognition, it can often be too weak of a signal to reliably detect visual themes in unlabeled, unsegmented images. In particular, appearance alone can be insufficient for discovery in the face of occluded objects, large intracategory variations, or low-resolution data.

In this work, we propose to discover novel categories that occur amid *known* objects within unannotated images. How could visual discovery benefit from familiar objects? The idea is that the relative layout of familiar visual objects surrounding less familiar image regions can help to detect patterns whose correct grouping may be too ambiguous if relying on appearance alone (see Fig. 1). Specifically, we propose to model the interaction between a set of detected categories and the unknown to-be-discovered categories, and show how a grouping algorithm can yield more accurate discovery if it exploits both object-level context cues as well as appearance descriptors.

As the toy example in Fig. 1 illustrates, novel recurring visual patterns ought to be more reliably detected in the presence of familiar objects. While both Figs. 1a and 1b contain multiple unknown objects, the common ones are difficult to isolate in Fig. 1a, whereas they are more quickly apparent in Fig. 1b once we realize that the known objects (circles, squares, and triangles) serve as "landmarks" for the unfamiliar one. We can infer that this new object is commonly found below squares and circles and above triangles, and that it itself has a certain shape/appearance structure.

Studies in perception confirm that humans use contextual cues from familiar objects to learn entirely new categories [6]. The use of familiar things as context applies even for nonvision tasks. For example, take natural language learning: When we encounter unfamiliar words, their definition can often be inferred using the contextual meaning of the surrounding text [7].

To implement this idea, we introduce a context-aware discovery algorithm. Our method first learns category models for some set of known categories. Given a new set

- *Y.J. Lee is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, ACES 3.302, 1 University Station C0803, Austin, TX 78712. E-mail: yjlee0222@utexas.edu.*
- *K. Grauman is with the Department of Computer Science, The University of Texas at Austin, 1616 Guadalupe, Suite 2.408, Austin, TX 78701. E-mail: grauman@cs.utexas.edu.*

Manuscript received 13 Dec. 2010; revised 12 Apr. 2011; accepted 8 May 2011; published online 6 June 2011.
Recommended for acceptance by G. Mori.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2010-12-0951.
Digital Object Identifier no. 10.1109/TPAMI.2011.122.

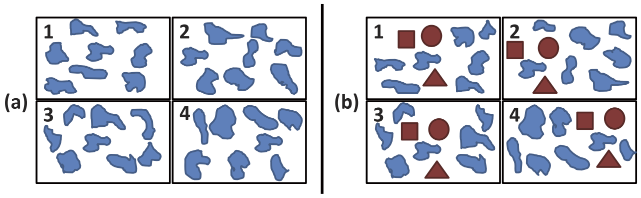


Fig. 1. Toy example giving the intuition for context-aware discovery. First cover (b) and try to discover the common object(s) that appear in the images for (a). Then, look at (b) and do the same. (*Hint: The new object resembles an* "r.") (a) When all regions in the unlabeled image collection are unfamiliar, the discovery task can be daunting; appearance patterns alone may be insufficient. (b) However, the novel visual patterns become more evident if we can leverage their relationship to things that are familiar (i.e., the circles, squares, and triangles). We propose to discover visual categories within unlabeled natural images by modeling interactions between the unfamiliar regions and familiar objects.

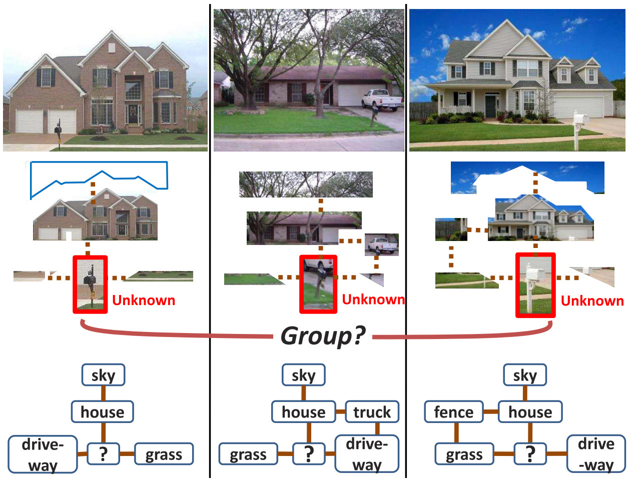


Fig. 2. Motivation for the proposed object-graph descriptor. We want to encode the layout of known categories relative to an unknown object. In this example, the unknown region is the *mailbox*. Our goal is to form clusters on the basis of the similarity of the unknown regions' appearance as well as the similarity between the graphs implied by surrounding familiar objects.

of completely unlabeled images, it predicts occurrences of the known classes in each image (if any), and then uses those predictions as well as the image features to mine for common visual patterns. For each image in the unlabeled input set, we generate multiple segmentations in order to obtain a pool of regions likely to contain some full objects. We classify each region as known (if it belongs to one of the learned categories) or unknown (if it does not strongly support any of the category models). We then group the unknown regions based on their appearance similarity and their relationship to the surrounding known regions. To model the intercategory interactions, we propose a novel *object-graph* descriptor that encodes the layout of the predicted classes (see Fig. 2). The output of the method is a set of discovered categories—that is, a partitioning of the unfamiliar regions into coherent groups.

The proposed method strikes a useful balance between current recognition strategies at either end of the supervision spectrum. The norm for supervised image labeling methods is forced-choice classification, with the assumption that the training and test sets are comprised of objects from the same pool of categories. On the other hand, the norm for unsupervised recognition is to mine for all possible categories from scratch [2], [3], [1], [4], [5]. In most real settings, we cannot predefine all categories of interest. For example, we cannot prescribe training data for all categories that a robot might encounter when navigating a new environment. The robot should be able to detect instances of the familiar objects for which it has training data, but should also be able to *discover* novel, unfamiliar objects.

In our approach, the system need not know how to label every image region, but instead can draw on useful cues from familiar objects to better detect novel ones. Ultimately, we envision a system that would continually expand its set of known categories—alternating between detecting what's familiar, mining among what's not, and then presenting discovered clusters to an annotator who can choose to feed the samples back as additional labeled data for new or existing categories.

Our setting of having partially labeled data is different from the traditional semi-supervised scenario. In semi-supervised learning, there are labeled and unlabeled data, but all instances are assumed to belong to the same set of categories. In our setting, instances in the unlabeled data belong to *disjoint* categories: some correspond to familiar categories, and some correspond to novel categories for which we have seen no prior training examples. Furthermore, whereas traditional semi-supervised learning treats labeled and unlabeled data in the same feature space, we use the familiar objects to describe the surrounding object-level context of the unfamiliar objects.

Our main contribution is the idea of context-aware unsupervised visual discovery. Our technique introduces 1) a method to determine whether regions from multiple segmentations are known or unknown, as well as 2) a new object-graph descriptor to encode object-level context. Unlike existing approaches, our method allows the interaction between known and unknown objects to influence the discovery. We evaluate our approach on the MSRC-v0, MSRC-v2, Corel, PASCAL 2008, and Gould 2009 [8] data sets and show that it leads to significant improvements in category discovery compared to strictly appearance-based baselines.

This paper expands upon our previous conference paper [9].

2 Related Work

In this section, we briefly review relevant work in unsupervised category discovery, the use of context for supervised object recognition, and 3D scene geometry estimates from single view data.

Existing unsupervised methods analyze appearance to discover object categories, often using bag-of-words representations and local patch features. Some methods leverage topic models, such as Latent Semantic Analysis or Latent Dirichlet Allocation, to discover visual themes [2], [3]. Others partition the image collection using spectral clustering [1], [4], [5], identify good exemplars with affinity propagation [10], or detect common patterns via local feature matches [11]. Our motivation is similar to these methods: to decompose large unannotated image collections into their common visual patterns or categories. However, while all previous methods assume no prior knowledge, the proposed approach allows intercategory interaction between familiar and unfamiliar regions to influence the groupings. This permits the discovery of objects that have similar appearance *and* similar surroundings, and may be inadequately clustered using appearance alone.

The idea of transferring knowledge obtained from one domain to a disjoint but similar domain is explored for object recognition in [12], [13]; the authors devise a prior based on previously learned categories, thereby learning with fewer labeled examples. In contrast, we directly model the interaction *between* the learned objects and the unknown to-be-discovered objects, thereby obtaining more reliable groups from unlabeled examples.

For supervised methods that learn from labeled images, several types of context have been proposed. Global image features [14] help to model the relationship between objects and scenes. Spatial context in the form of neighboring region information can be modeled with pairwise relations [15] and with interpixel or interregion spatial interactions [16], [17], [18] or top-down constraints [19]. The benefit of high-level semantic context based on objects' co-occurrence and relative locations is demonstrated in [20], [21]. Without such information, impoverished appearance (e.g., due to low resolution) can severely hurt recognition accuracy [22].

Our method exploits high-level semantic context for unsupervised category discovery. Unlike the above supervised methods, we do not learn about intercategory interactions from a labeled training set, nor do we aim to improve the detection of familiar objects via context relationships. Instead, we identify contextual information in a data-driven manner by detecting patterns in the relative layout of known and unknown object regions within unlabeled images. The method in [23] recovers contextual information on-the-fly from the test images by exploiting the data's statistical redundancy. However, in contrast to our approach, that method learns the context surrounding familiar object instances to improve their classification, whereas our approach discovers object-level context surrounding *unfamiliar* object regions to improve their grouping (discovery of new objects).

Recent approaches analyze 3D scene geometry for recognition tasks. The 3D scene context of an image is used to model the interdependence of objects, surface orientations, and camera viewpoint for object detection [24] and region labeling [8]. A recent method performs occlusion reasoning to recover approximate depth orderings of objects [25]. Another method uses a Markov Random Field (MRF) to model the 3D reconstruction of an image as a function of the image features and the relations between depths at various points in the image [26]. Whereas these methods aim to improve supervised classification accuracy, we explore how depth cues can be utilized to enhance 3D object-level context for discovery.

3 Approach

The goal is to discover categories in unlabeled image collections using appearance and object-level semantic context cues. Our approach first acquires knowledge from a set of labeled "known" category examples and builds classifiers for each class. Then, given a new collection of unlabeled data, we segment each image into coherent regions. To increase the likelihood of obtaining some regions that correspond to true objects, we work with multiple segmentations. We classify each region as "known" or "unknown" depending on the confidence that the region belongs to one of the learned categories. For each unknown region, we represent its interaction with surrounding known objects via the proposed object graph, which encodes both the class distributions and their relative displacement. Finally, we group together regions from all images that have similar appearance *and* object graphs.

What are the main assumptions of our approach? For any object to be discovered, its visual pattern must be recurring and its surrounding familiar objects should belong to a similar set of categories and share similar configurations across the image collection. This means that co-occurring objects that are often segmented together, such as bicycles and bicycle racks, can be discovered as a single category.[1] In addition, we assume that each object in the image will have a corresponding segment that roughly agrees with its true boundaries among the multiple segmentations. This is a common premise when working with multiple segmentations [27], [2], and we validate it directly in Section 4.9.

3.1 Identifying Unknown Objects

Any image in the unlabeled collection may contain multiple objects and may have a mixture of familiar and unfamiliar regions. In order to describe the interaction of known and unknown objects, first we must predict which regions are likely instances of the previously learned categories. The problem of distinguishing known regions from unknown regions has not directly been addressed in the recognition literature, to our knowledge, as most methods aim to either classify the image as a whole, label every pixel with a category, or localize a particular object.

Ideally, an image would first be segmented such that each region corresponds to an object; then we could classify each region and take only those with the most confident outputs as "knowns." In practice, due to the nonhomogeneity of many objects' appearance, bottom-up segmentation algorithms (e.g., Normalized Cuts [28]) cannot produce such complete regions. Therefore, following [2], [29], we generate *multiple segmentations* per image, with the expectation that although some regions will fail to agree with object boundaries, some will be good segments that correspond to coherent objects. Each segmentation is the result of varying the parameters to the segmentation algorithm (i.e., number of regions, image scale). As in previous work, each segment goes into the pool of instances that will be processed by the algorithm, which means segments that overlap in the same original image are treated as separate instances.

We first compute the confidence that any of these regions correspond to a previously learned category. Assuming reliable classifiers, we will see the highest certainty for the "good" regions that are from known objects, and lower

[1] Note, however, that in our experiments, we evaluate discovery given human labeled categories, in which case bicycles and bicycle racks are treated as separate categories, i.e., we will be penalized for grouping them together.

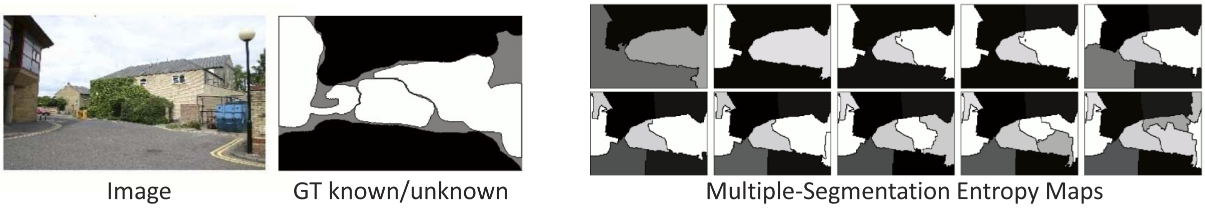


Fig. 3. An example image, its ground-truth known/unknown label image, and our method's predicted entropy maps for each of its 10 segmentations. For the ground truth, black regions denote **known** classes (sky, road) and white regions denote **unknown** classes (building, tree). (Gray pixels are "void" regions that were not labeled in the MSRC-v2 ground truth.) In the entropy maps, lighter/darker colors indicate higher/lower entropy, which signals higher/lower uncertainty according to the known category models. Note that the regions with the highest uncertainty (whitest) correspond correctly to unknown objects, while those with the lowest uncertainty (darkest) are known. Regions that are comprised of both known and unknown objects are typically scored in between (gray). By considering confidence rates among multiple segmentations, we can identify the regions that are least strongly "claimed" by any known model.

responses on regions containing a mix of known and unknown objects or regions composed entirely of unknown objects (see Fig. 3). Using this information to sort the regions, we can then determine which need to be sent to the grouping stage as candidate unknowns and which should be used to construct the surrounding object-level cues.

We use a labeled training set to learn classifiers for N categories, $C = \{c_1, \dots, c_N\}$. The classifiers must accept an image region as input and provide a confidence of class membership as output. We combine texture, color, and shape features using the multiple kernel learning (MKL) framework in [30] and obtain posterior probabilities for any region with an SVM classifier; i.e., the probability that a segment s belongs to class c_i, $\mathrm{P}(c_i|s)$. (Details on the features we use in our results are given in Section 4.)

The familiarity of a region is captured by the list of these posterior probabilities for each class, which reflects the class-label confidences given the region. Segments that look like a learned category c_i will have a high value for $\mathrm{P}(c_i|s)$, and low values for $\mathrm{P}(c_j|s)$, $\forall j \neq i$. These are the known objects. Unknown objects will have more evenly distributed values or multiple peaks among the posteriors. Thus, to measure the degree of uncertainty, we compute the entropy E for a segment s, $E(s) = -\sum_{i=1}^{N} \mathrm{P}(c_i|s) \cdot \log_2 \mathrm{P}(c_i|s)$.

The lower the entropy, the higher the confidence that the segment belongs to one of the known categories. Similarly, higher entropy regions have higher uncertainty and are thus more "unknown." This gives us a means to separate the known regions from the unknown regions in each image (see Fig. 4). Note that entropy ranges from 0 to $\log_2(N)$; we simply select a cutoff threshold equal to the midpoint in this range, and treat regions above the threshold as unknown and those below as known. Fig. 3 shows the entropy maps we computed for the multiple segmentations from a representative example image. Note the agreement between the highest uncertainty ratings and the true object boundaries.

3.2 Object Graphs: Modeling the Topology of Category Predictions

Given the unknown regions identified above, we would like to model their surrounding contextual information in the form of object interactions. Specifically, we want to build a graph that encodes the topology of adjacent regions relative to an unknown region (recall the mailbox example in Fig. 2). Save for the unknown regions, the nodes are named objects and edges connect adjacent objects. With this representation, one could then match any two such graphs to determine how well the object-level context agrees for two candidate regions that might be grouped. Regions with similar surrounding context would have similar graphs; those with dissimilar context would generate dissimilar graphs.

If we could rely on perfect segmentation, classification, *and* separation of known and unknown regions, this is exactly the kind of graph we would construct—we could simply count the number and type of known objects and record their relative layout. In practice, we are limited by the accuracy and confidence values produced by our classifier as well as the possible segments. While we cannot rectify mislabeled known/unknown regions, we can be more robust to misclassified known regions (e.g., sky that could almost look like water) by incorporating the uncertainty into the surrounding object context description.

To that end, we propose an *object-graph* descriptor that encodes the likely categories within the neighboring segments and their proximity to the unknown base segment. Rather than form nodes solely based on a region's class label with the maximum posterior probability, we create a histogram that forms localized counts of object presence weighted according to each class's posterior. For each segment, we compute a distribution that averages the probability values of each known class that occurs within that segment's r spatially nearest neighboring segments (where nearness is measured by distance between segment centroids), incremented over increasing values of r (see Fig. 5).

Specifically, for each unknown segment s, we compute a series of histograms using the posteriors computed within its neighboring superpixels. Each component histogram

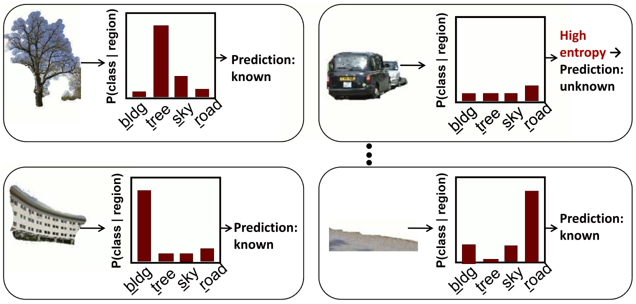


Fig. 4. Familiar versus unfamiliar object predictions. For all segments, we use classifiers to compute posteriors for the N known categories. We treat each segment as either known or unknown based on the resulting entropy score. Here and in our running example, there are four known classes: building, tree, sky, and road.

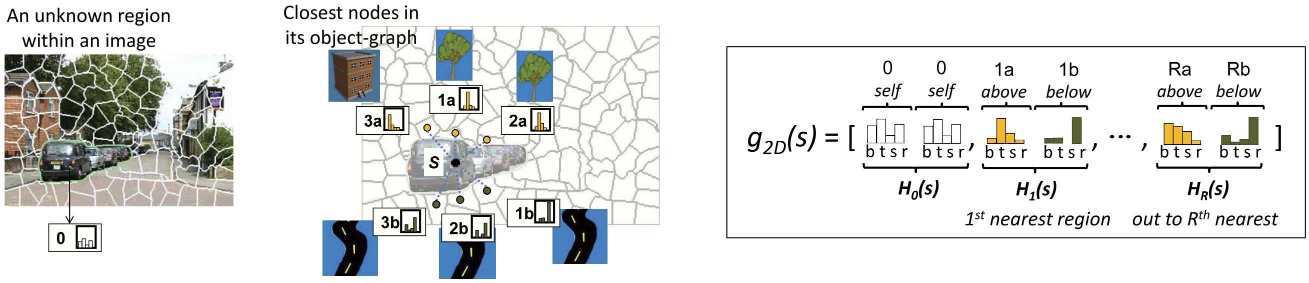


Fig. 5. Schematic of the proposed 2D object-graph descriptor. The base segment is s. The numbers indicate each region's rank order of spatial proximity to s for two orientations, *above* and *below*. The circles denote each segment's centroid. In this example, there are four known classes: building (**b**), tree (**t**), sky (**s**), and road (**r**). Each histogram $H_r(s)$ encodes the average posteriors for the r neighboring segments surrounding s from above or below, where $0 \leq r \leq R$. (Here, $R = 3$ and bars denote posterior values.) Taken together, $g_{2D}(s)$ serves as a soft encoding of the likely classes that occur relative to s, from near to far, and at two orientations.

$H_r(s)$ accumulates the average probability of occurrences of each class type c_i within s's r spatially nearest segments for each of two orientations, *above* and *below* the segment. We retain the superpixel segment centered on the unknown segment and remove the remaining segments that overlap with the unknown segment. We concatenate the component histograms for $r = 0, \ldots, R$ to produce the final object-graph descriptor:

$$g_{2D}(s) = [H_0(s), H_1(s), \ldots, H_R(s)], \tag{1}$$

where $H_0(s)$ contains the posteriors computed within s's central superpixel. The result is an $((R+1) \cdot 2N)$-dimensional vector, where N denotes the number of familiar classes. Note that higher values of r produce a component $H_r(s)$ covering a larger region, and the descriptor softly encodes the surrounding objects present in increasingly further spatial extents. Our representation can detect partial context matches (i.e., partially agreeing spatial layouts) since the matching score between two regions is proportional to how much their context agrees. Due to the cumulative construction, discrepancies in more distant regions have less influence.

There are a couple of implementation details that will help ensure that similar object topologies produce similar object-graph descriptors. First, we need to maintain consistency in the size and relative displacement of nodes (regions) across different object graphs; to do this, we use superpixel segments as nodes (typically about 50 per image). Their fairly regular size and shape tessellates the image surrounding the unknown region well, which in turn makes a centroid-based distance between nodes reliable.[2] As usual, the superpixels may break nonhomogeneous objects into multiple regions, but as long as the over-segmentation effect is fairly consistent in different images (e.g., the dark roof and light wall on the building are often in different superpixels), the object graph will avoid misleading double-counting effects. Empirically, we have observed that this consistency holds.

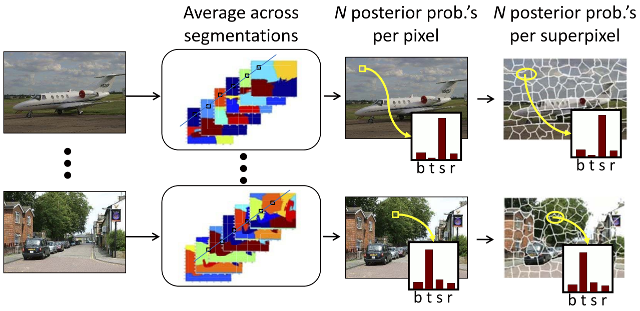

Average across segmentations
N posterior prob.'s per pixel
N posterior prob.'s per superpixel
b t s r
b t s r
b t s r
b t s r

Fig. 6. Superpixel summarization of object predictions. We map the per-region posteriors to per-pixel posteriors by averaging values pixel-wise across all segmentations computed for a given image. Superpixel regions are then assigned posteriors using the average of their member pixels' posteriors. This allows us to estimate the known classes' presence from larger regions, but then summarize the results in the smaller superpixel nodes.

Second, we need to obtain robust estimates of the known objects' posterior probabilities and avoid predicting class memberships on regions that are too local (small). For this, we exploit the multiple segmentations: We estimate the class posteriors for each segment, then for each image, we stack its segmentation maps and compute a per-pixel average for each of the N posterior probabilities. Finally, we compute the posteriors for each superpixel node by averaging the N-vector of probabilities attached to each of its pixels (see Fig. 6). Note that this allows us to estimate the known classes' presence from larger regions, but then summarize the results in the smaller superpixel nodes.

We select a value of R large enough to typically include all surrounding regions in the image. We limit the orientations to above and below (as opposed to also using left and right) since we expect this relative placement to have more semantic significance; objects that appear side by side can often be interchanged from left to right (e.g., see the mailbox example in Fig. 2). For images that contain multiple unknown objects, we do not exclude the class-probability distributions of the unknown regions present in another unknown region's object graph. Even though the probabilities are specific to known objects, their distributions still give weak information about the appearance of unknown objects. The probabilities cannot denote which class the unknown region should belong to (since all possible answers would be incorrect), but we will get similar distributions for similar-looking unknown regions. As long as the unknown objects consistently appear in similar surrounding displacements throughout the data set (e.g.,

[2] Note that our descriptor assumes images have similar scene depth, and thus that the relative placement of surrounding objects depends only on the scale of the object under consideration (as do most existing recognition methods using object co-occurrence context, e.g., [16], [17]). In Section 3.3, we relax this assumption to encode a 3D object-graph descriptor that utilizes scene depth.

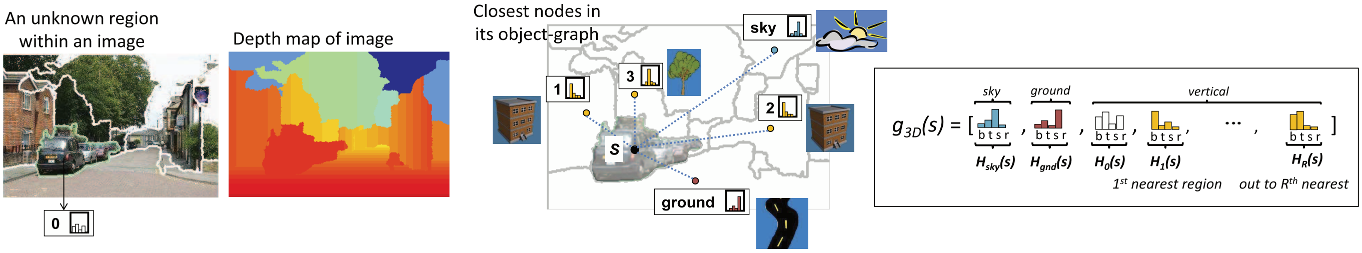


Fig. 7. Schematic of the 3D object-graph descriptor. The base segment is s. The numbers indicate each region's rank order of estimated depth proximity to $\boldsymbol{s}$. We compute the depth map using the method of Hoiem et al. [25]. $H_{sky}(s)$ and $H_{gnd}(s)$ encode the posteriors for the sky-plane and ground-plane segments, respectively. Each $H_r(s)$ encodes the average posteriors for the r neighboring vertical-plane segments surrounding s, where $0 \leq r \leq R$. Taken together, the object graph $g_{3D}(s)$ serves as a soft encoding of the likely classes that occur relative to s, from near to far in terms of scene depth, and at three surface orientations.

unfamiliar cows appearing near other unfamiliar cows), it should only aid the contextual description.

Previous methods have been proposed to encode the appearance of nearby regions or patches [16], [17], [31], [5]; however, our object graph is unique in that it describes the region neighborhood based on object-level information, and explicitly reflects the layout of previously learned categories. (In Section 4, we demonstrate the comparative value for the discovery task.) Relative to existing graph kernels from the machine learning literature [32], [33], our approach allows us to represent object topology without requiring hard decisions on object names and idealized segmentations.

3.3 Three-Dimensional Object Graphs

In this section, we show how to extend our object-graph descriptor to model 3D spatial layout.

The 2D object graph is often sufficient to model the scene context and relative locations of objects since general photographer biases lead to similar 2D layouts across image instances (e.g., sky is above, ground is below, and camera distance to objects is within a similar range). However, in some cases, the spatial relationships between objects in the 2D image plane can appear to be quite different from their true relationships in the 3D world. Explicitly modeling the 3D scene geometry can resolve potential discrepancies in spatial relationships between objects in images with different scene depths. (For example, in a close-up photo of a car, a part of the road that is actually behind the car would be placed *above* the car in the 2D image plane. By modeling scene geometry, we can infer that the road is actually *below* the car in the 3D world plane, and thus make its scene context comparable to that of a car in a broader street scene image.) Thus, to account more explicitly for the depth ordering of objects in the scene, we next introduce a 3D variant of the object graph that uses single-view estimates of occluding boundaries to estimate the proximity and relative orientations of surrounding familiar objects.

Given a depth ordering of the objects in the image, the 2D object-graph descriptor can be adapted to capture the relationships between the objects in the 3D world. To estimate depth, we employ the method of Hoiem et al. [25], which infers occlusion boundaries in a single image using edge and region cues together with 3D surface and depth cues. It computes a segmentation of the image, classifies each region as belonging to either the *sky*, *ground*, or *vertical* planes, and produces pixel-level depth estimates. We compute a single depth estimate for each region by averaging its pixel-level depth values.

To create our 3D object-graph descriptor, we encode the likely categories within the neighboring segments and their proximity to the unknown base segment with cumulative posterior probability histograms. Unlike the 2D object-graph descriptor, which ranks neighboring regions based on their centroid distances in the image plane, the 3D object-graph descriptor measures region nearness using 3D depth estimates, explicitly accounting for the surface planes (e.g., *sky*, *ground*, and *vertical*) that each region resides in. Furthermore, we use regions rather than superpixels for the 3D object-graph nodes since 1) the regions generated using [25] cover objects quite well, and 2) we no longer assume similar scene depth across images and thus do not benefit from the superpixels' consistency in size and relative displacements. Instead, for each surface plane, we accumulate the posterior probability distributions of neighbors in increasing displacement in depth (as measured by L2 distance) relative to the central unknown object. We then concatenate the posterior distributions to create a single 3D object-graph descriptor for the unknown region:

$$g_{3D}(s) = [H_{sky}(s), H_{ground}(s), H_0(s), \ldots, H_R(s)]. \qquad (2)$$

Fig. 7 shows a schematic of the 3D object-graph descriptor.

In Section 4.11, we compare the 2D and 3D object-graph variants. The performance of the 3D object graph is influenced by the accuracy of the underlying scene depth estimate algorithm. In our experiments, we observe that the method of [25] produces best results for scene images with multiple objects (including sky and ground) and a visible horizon, and it is less reliable for images of close-up objects. While we focus on single-view estimates of relative depth to avoid making assumptions about the original sensor, of course if stereo data were available our method could similarly exploit it.

3.4 Category Discovery amid Familiar Objects

Now that we have a means to compute object-level context, we can combine this information with region-based appearance to form homogeneous groups from our collection of unknown regions. We define a similarity function between two regions s_m and s_n that includes both region appearance and known-object context:

$$K(s_m, s_n) = \frac{1}{|\mathrm{u}|} \sum_{\mathrm{u}} K_{\chi^2}(a_{\mathrm{u}}(s_m), a_{\mathrm{u}}(s_n)) + K_{\chi^2}(g(s_m), g(s_n)),$$

where $g(s_m)$ and $g(s_n)$ are the object-graph descriptors (either of the 2D or 3D variants), and each $a_{\mathrm{u}}(s_m)$ and $a_{\mathrm{u}}(s_n)$ denotes

Fig. 8. Example images of the data sets used in our experiments.

an appearance-based feature histogram extracted from the respective region (which will be defined in Section 4). Each $K_{\chi^2}(\cdot,\cdot)$ denotes a χ^2 kernel function for two histogram inputs: $K_{\chi^2}(x,y) = \exp(-\frac{1}{2}\sum_i \frac{(x_i-y_i)^2}{x_i+y_i})$, where i indexes the histogram bins.

We compute affinities between all pairs of unknown regions to generate an affinity matrix, which is then given as input to a clustering algorithm to group the regions. We use the spectral clustering method developed in [34]. Because we use multiple segmentations, if at least one "good" segment of an unknown object comes out of an image, then it may be matched and clustered with others that belong to the same category. Since our unknown/known separation for novel images may be imperfect, some discovered groups may contain objects that actually belong to a known class. Importantly, since affinity can be boosted by either similar appearance *or* similar context of known objects, we expect to be able to discover objects with more diverse appearance.

We summarize the steps of our algorithm in Algorithm 1. We provide source code and data annotations at the project webpage: http://vision.cs.utexas.edu/projects/objectgraph/.

Algorithm 1. The context-aware discovery algorithm.

```
Input: Set of classifiers for $N$ known category models,
       set of novel unlabeled images, and $k$.
Output: Set of $k$ discovered categories (clusters).
1. Obtain multiple segmentations for each image.
2. Compute posteriors for each region. (Section 3.1)
3. Compute the entropy for each region to classify as
   "known" or "unknown". (Section 3.1)
4. Construct an object graph for each unknown region.
   (Sections 3.2 and 3.3)
5. Compute affinities between unknown regions with the
   object graph and appearance features, and cluster to
   discover categories. (Section 3.4)
```

4 Results

In this section, we

1. evaluate our method's discovery performance and compare against two appearance-only baselines,
2. analyze our entropy-based known-unknown separation measure,
3. compare the object graph with an appearance-based context baseline,
4. compare the 2D and 3D object-graph variants, and
5. show qualitative examples of real object graphs and discovered categories.

4.1 Data Sets

We validate our approach with five data sets: MSRC-v0, MSRC-v2, PASCAL VOC 2008, Corel, and Gould 2009 [8]. Fig. 8 shows examples of each data set. Our data set selection is based on the requirements that we have images with pixel-level ground truth and multiple objects from multiple categories. The Gould 2009 data set is chosen to test our 3D object-graph performance as it has been tested for computing depth estimates previously, in [8]. To our knowledge, these are the best and most recent sets satisfying these requirements.

The original MSRC-v0 contains only weakly labeled images, so we used Mechanical Turk to obtain pixel-level ground truth for all the images with multiple objects (3,457 images total), and created 21 classes. MSRC-v2 contains 21 classes and 591 images, PASCAL contains 20 classes and 1,023 images (we use the trainval set from the segmentation tester challenge), Corel contains seven classes and 100 images, and Gould 2009 contains 14 classes and 715 images. (The original Gould 2009 data set contains eight classes that include a generic "foreground" class. We annotated foreground class regions with one of seven additional, more specific labels: *car, bus, boat, cow, sheep, motorbike, person*.) We evaluate all sets for accuracy, and focus additional analysis on MSRC-v2 since it has the largest number of categories and ground-truth labeling [29] for all objects in the data set.

We want to evaluate how sensitive our method is with respect to which classes are considered familiar (or unfamiliar) and how many (or few) objects are in the "known" set of models. Thus, for each data set, we form multiple splits of known/unknown classes for multiple settings of both the number of knowns (N) and the number of true unknowns (U) present. We learn the known classes on 60 percent of the data and run our discovery algorithm on the other 40 percent.

For MSRC-v2, we create two sets for each of three different split sizes: $U = [5, 10, 15]$, $N = [16, 11, 6]$, forming six total variations. (The U and N denote the number of unknown and known categories, respectively.) Similarly, for PASCAL, we create two sets each for three sizes: $U = [5, 10, 15]$, $N = [15, 10, 5]$. For the smaller Corel set, we create a single split with $U = 2$ and $N = 5$. For MSRC-v0, we create a single split with $U = 8$ and $N = 13$. For Gould 2009, we create a single split with $U = 7$ and $N = 7$. For Corel, MSRC-v0, and Gould 2009, we choose the split manually, selecting as unknown those categories that we think could benefit most from object-level context. However, for

MSRC-v2 and PASCAL, we select all 12 splits randomly. See Table 1 in the Appendix for a detailed breakdown of the category names in each split, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.122.

4.2 Implementation Details

Our implementation choices apply to all data sets and experiments, unless specified otherwise.

We use Normalized Cuts [28] to generate multiple segmentations for each image by varying the number of segments from 3 to 12 and applying these settings at image scale 150 pixels across, similar to [2]. This results in 10 segmentations (75 segments) per image. To form the appearance descriptor $a_u(s)$ for a region s, we use several types of bag-of-features histograms: Texton Histograms (TH), Color Histograms (CH), and pyramid of HOG (pHOG) [35]. To compute TH, we use a filter bank with 18 bar and 18 edge filters (six orientations and three scales for each), one Gaussian filter, and one Laplacian-of-Gaussian filter. These responses are quantized to 400 textons via k-means. We compute the CH features in Lab color space, with 23 bins per channel. We compute pHOG features with three pyramid levels with eight bins, which results in a 680-dimensional descriptor. We normalize each $a_u(s)$, $g_{2D}(s)$, and $g_{3D}(s)$ to sum to 1. We train our known classifiers using ground-truth segmentations. To compute class probabilities for the regions, we use one-versus-all SVM classifiers trained using MKL, and obtain posteriors using [36]. For our 2D object-graph descriptor, we generate an oversegmentation containing roughly 50 superpixels for each image and fix the neighborhood range at $R = 20$ per orientation.

Due to the large number of images in the MSRC-v0, performing discovery with spectral clustering can be very time consuming. Therefore, we add a pruning step prior to clustering, where for each image, we retain the most unknown regions (i.e., those that had highest entropy) and remove all overlapping regions.

4.3 Evaluation Metrics

We use both *purity* [37] and *mean Average Precision* (mAP) to quantify accuracy. The former rates the coherency of the clusters discovered, while the latter reflects how well we have captured the affinities between intraclass versus interclass instances (independent of the clustering algorithm). We only consider regions with ground-truth labels (i.e., no "voids" from MSRC). To score an arbitrary segment, we consider its ground-truth label to be that to which the majority of its pixels belong.

These metrics reward discovery of object parts as well as full objects (e.g., we would get credit for discovering cow heads and cow legs as separate entities). This seems reasonable for the unsupervised category discovery problem setting, given that the part/object division is inherently ambiguous without external human supervision. We evaluate all methods across different settings of the number of discovered objects, k. In this way we intend to see to what extent our method's advantages are stable with respect to the granularity of the discoveries. Since the spectral clustering step [34] uses a random initialization, we average all results over 10 runs.

4.4 Unsupervised Discovery Accuracy

To support our claim that the detection of familiar objects should aid in category discovery, we evaluate how much accuracy improves when we form groups using appearance together with the object graph versus when we form groups using appearance alone. We thus generate two separate curves for purity scores: 1) an appearance-only baseline where we cluster unknown regions using only appearance features (App. only), and 2) our approach where we cluster using both appearance and contextual information (Object Graph).

Since our evaluation scenario necessarily differs from earlier work in unsupervised discovery, it is not possible to directly compare the output of our method with previously reported numbers: Our method assumes some background knowledge about a subset of the classes, whereas existing discovery methods assume none. However, our appearance-only baseline is intended to show the limits of what can be discovered using conventional approaches for these data, since previous unsupervised methods all rely solely on appearance [2], [1], [4], [5]. In all results, our method and the baseline are applied to the same pool of segments (i.e., those our method identifies as unknown).

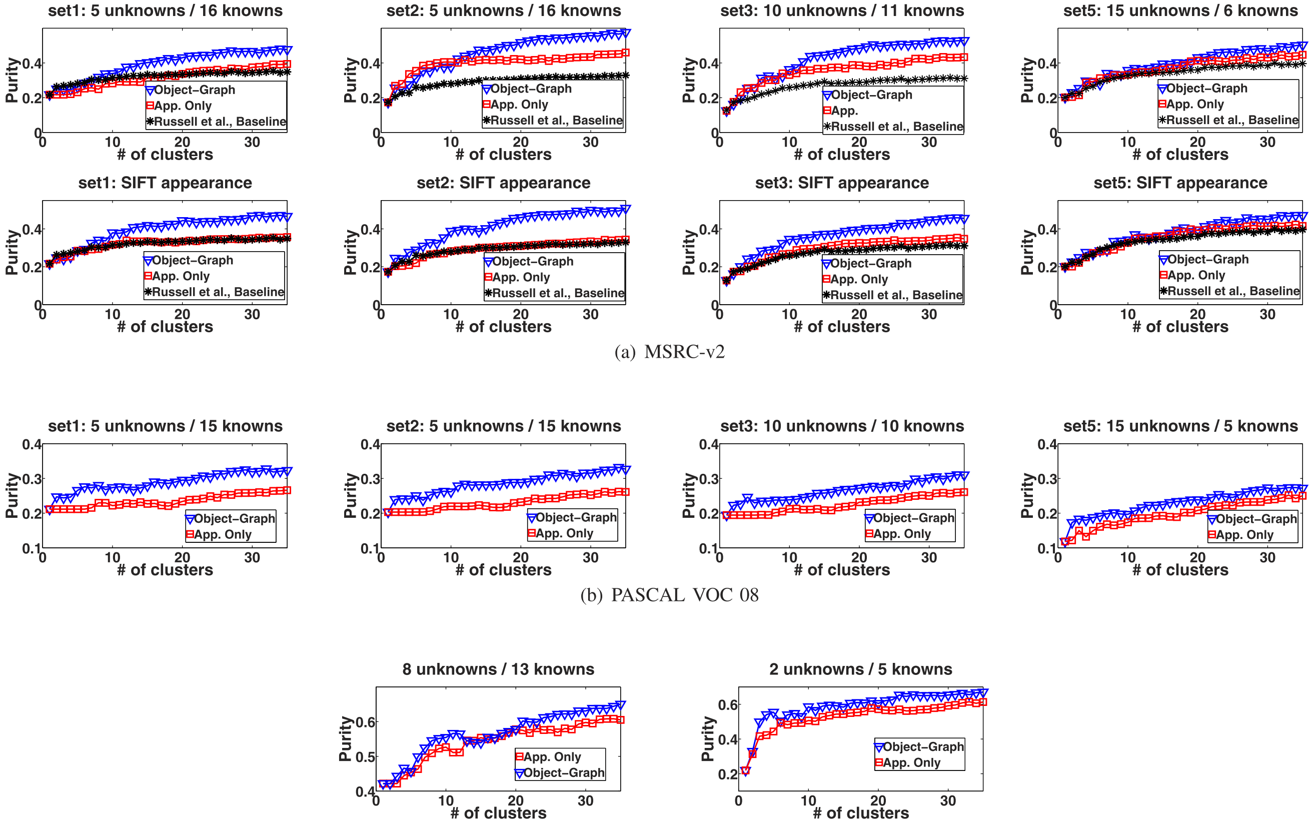
Figs. 9a, 9b, 9c, and 9d show the results using the 2D object graph on four data sets. Our model significantly outperforms the appearance-only baseline. These results confirm that the appearance and object-level contextual information complement each other to produce high quality clusters. Figs. 9a and 9b illustrate our method's consistency with respect to various random splits of unknown/known category pools, and will be discussed below.

To directly evaluate how accurately our object-graph affinities compare the regions, we can remove the clustering step and analyze the mean Average Precision (see Table 1). Our full model noticeably outperforms the appearance-only baseline in all categories. In fact, the object-graph descriptor alone (with no appearance information) performs almost as well as our full model. For bicycles, the affinities obtained using only appearance information are weak, and thus the full model actually performs slightly worse than the object-graph descriptor in isolation. We also see that our model's largest improvement occurs for the cow class (high appearance variance), whereas it is smaller for trees (low appearance variance). This makes sense because context is more helpful when grouping instances from a category with high appearance variation.

4.5 Comparison to the State of the Art

We next generate comparisons with the state-of-the-art Latent Dirichlet Allocation (LDA)-based discovery method of Russell et al. [2] using the authors' publicly available code. For this baseline, we use a Bag-of-Features representation with SIFT features (as proposed in [2]). To our knowledge, theirs is the only other current unsupervised method that tests with data sets containing multiple objects per image, making it the most suitable method for comparison. As before, our method and the baseline are applied to the same pool of segments.

The plots in the first row in Fig. 9a show the results on the MSRC-v2. Our full model significantly outperforms the LDA baseline, which corroborates the result from the



(a) MSRC-v2

(b) PASCAL VOC 08

(c) MSRC-v0

(d) Corel

Fig. 9. Discovery accuracy results given by purity rates for all four 2D object-graph data sets as a function of k. Higher curves are better. We compare our 2D object-graph approach (Object Graph) with appearance-only baselines. The discovered categories are more accurate using the proposed approach, as the familiar objects nearby help us to detect region similarity even when their appearance features may only partially agree. Note that for (a) and (b), there are fewer knowns in the splits from left to right.

previous section that modeling object-level contextual information leads to higher quality clusters than those attainable by using appearance information alone.

To ensure that the improvement over [2] is not a result of stronger appearance features, we repeated the experiment using the same features for all methods, letting $a_{\mathrm{u}}(s)$ be a SIFT bag of words as in [2]. The second row in Fig. 9a shows the results. Again, our method substantially outperforms the baseline, even though it performs slightly worse than when using TH, CH, and pHOG for appearance features. Note that the two appearance-based methods (black and red curves) show even closer results, which indicates that most of the improvement in accuracy can be attributed to the object-graph.

TABLE 1
Mean Average Precision Scores
for the Unknown Categories of the MSRC-v2 set1

	Building	Tree	Cow	Airplane	Bicycle
Our full model	**0.32**	**0.36**	**0.41**	**0.36**	0.21
App. only	0.27	0.33	0.20	0.21	0.10
Obj-Graph only	**0.32**	0.27	0.37	0.32	**0.24**

Our 2D object-graph method's largest improvement occurs for the cow class (high appearance variance), whereas it is smaller for trees (low appearance variance).

4.6 Impact of Known/Unknown Decisions

We next evaluate how accurately our model estimates true familiar versus unfamiliar regions. Fig. 10a shows the precision-recall curve for our known-unknown decisions on the MSRC-v2 set1. For this, we treat the known classes as positive, and the unknown classes as negative, and sort the regions by their entropy scores. The red star indicates the precision-recall value at $\frac{1}{2}\max E(s)$. With this (arbitrary) threshold, the regions considered for discovery are almost all true unknowns (and vice versa), at some expense of misclassifying unknown and known regions. Adjusting the "knob" on the threshold produces a trade-off between the number of true unknowns considered for discovery versus the number of true knowns treated as unknowns. Learning the optimal threshold depends on the application, and for our problem setting, $\frac{1}{2}\max E(s)$ suffices.

How much better could we do with more reliable predictions of what is unknown? Fig. 10b shows the results for the MSRC-v2 set1 if we replace our known-unknown predictions with perfect separation (note the vertical axis scale change). Again, our model outperforms the appearance-only baseline. All purity rates are notably higher here compared to when the known/unknown separation is computed automatically, likely because the discovery problem has become much simpler: Instead of having regions that could belong to one of 21 categories (total number of known and unknown categories), we only need to group the true unknowns. This implies that there is room

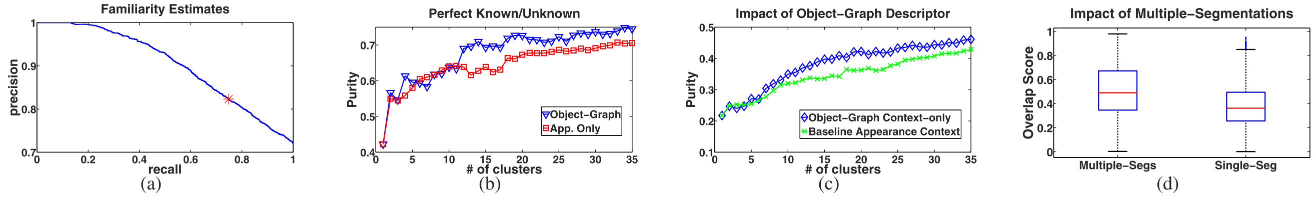


Fig. 10. Analysis of the known/unknown decisions, the 2D object-graph descriptor, and the bottom-up multiple segmentations. (a) Precision-recall curve for known versus unknown decisions; the star denotes the cutoff (half of the maximum possible entropy value). (b) Discovery accuracy using perfect known/unknown separation. Note the purity axis scale difference compared to Fig. 9a. (c) Comparison of the 2D object-graph descriptor to a "raw" appearance-based context descriptor. (d) Maximal segmentation accuracy attainable per object using multiple segmentations versus a single segmentation. These results are on the MSRC-v2 set1.

for better initial classification (i.e., better label predictions and confidences), with which we can expect higher cluster purity rates.

4.7 Impact of Which Categories Are Familiar

Upon examining the relative performance on different known/unknown splits, we find that discovery performance depends to a limited extent on which categories are known and how many. For example, both our method and the baseline have stronger discovery performance on MSRC-v2 set2 than on set1. This can be attributed to the fact that the unknowns in set2 are *grass, sky, water, road,* and *dog*, which have strong appearance features and can be discovered reliably without much contextual information. When the ratio between the number of unknown categories to known categories increases (from left to right in Figs. 9a and 9b), there is a decrease in the information provided by the known object-level context and, consequently, we find that our improvements over the baseline eventually have a smaller margin (see the rightmost curves in Figs. 9a and 9b, where only five or six objects are known). Overall, however, we find that the improvements are quite stable: Across the 12 random splits of variably sized sets of known/unknowns tested for the MSRC and PASCAL, our method never detracts from the accuracy of the appearance-only baseline.

4.8 Impact of the Object-Graph Descriptor

In this section, we evaluate how our 2D object-graph descriptor compares to a simpler alternative that directly encodes the surrounding appearance features. Since part of our descriptor's novelty rests on its use of object-level information, this is an important distinction to study empirically. We substitute class-probability counts in the object graph with raw feature histogram counts using concatenated TH, CH, and pHOG histograms.

Fig. 10c shows the result on the MSRC-v2 set1. Our object graph performs noticeably better than the baseline, confirming that directly modeling class interactions instead of surrounding appearance cues can improve discovery. Even though our initial classification results are based on the same information, learned category distributions are more reliable than local appearance patterns since they model high-level object representations as opposed to low-level texture or color information.

In addition to improved accuracy, our descriptor also has the advantage of lower dimensionality. The object graph requires only $R \cdot 2N$-dimensional vectors for each unknown region, whereas the appearance baseline requires $R \cdot 2Q$-dimensional vectors, for Q texton + color + pHOG bins. In this case, our object graph is about 70 times more compact.

4.9 Impact of Multiple Segmentations

We next study the impact that multiple segmentations have on providing candidate object regions that agree well with true boundaries. For each object in the image, we take the region from the pool of bottom-up multiple segmentations that has the highest overlap score with its ground-truth segmentation to compute the maximum overlap score [38]. We compare against taking regions from a single segmentation baseline that generates seven segments per image (the average number of regions per segmentation in the set of multiple segmentations).

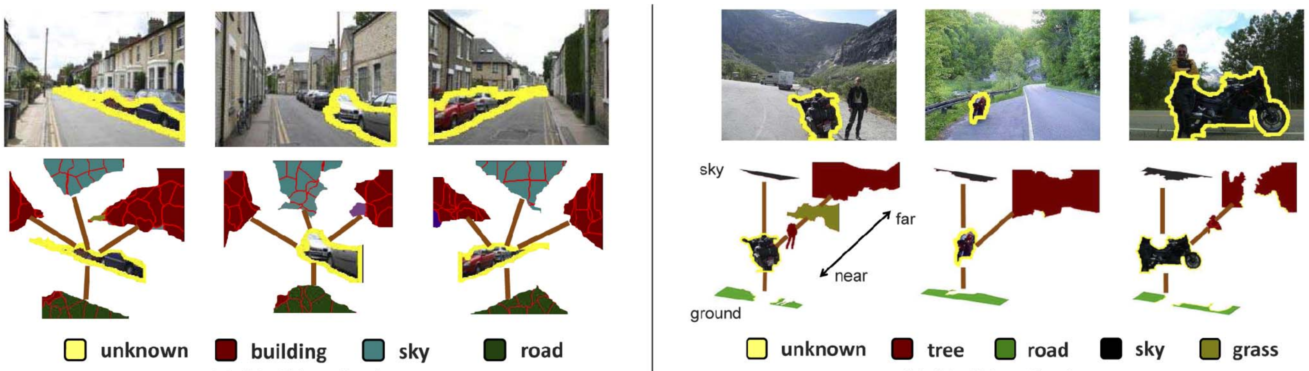
Fig. 10d shows the result on MSRC-v2. The regions in the pool of multiple segmentations provide significantly better candidates for representing objects than those in the pool of the single segmentation baseline. The median score for the multiple segmentation regions is about 0.5, which indicates that the best candidates have high overlap with true object regions. This result corroborates the findings in [2].

While the result highlights the importance of generating multiple segmentations, it also reveals the limitations of bottom-up segmentations for discovery, since there is clearly room for improvement in segmentation quality. In ongoing work [39], we are exploring how discovered top-down patterns in the unlabeled image collection can be used to refine the regions, so that we are not restricted to discovering patterns among the bottom-up segments.

4.10 Example Object Graphs

Figs. 11a and 11b show examples of 2D and 3D object graphs generated using our approach, respectively. The 2D object graphs are generated on the MSRC-v0 data set with *building, grass, sky, road, mountain, water, flower,* and *leaf* as knowns, and the 3D object graphs are generated on the Gould 2009 data set with *sky, tree, road, grass, water, building,* and *mountain* as knowns.

Our method correctly identifies the car and motorbike regions as unknowns (those with yellow boundaries) and produces accurate descriptions of the surrounding familiar object-level context. To visualize the familiar category posterior distributions in each surrounding region node, we label each node with the category that produces the *maximum* posterior probability. Furthermore, for the 2D object graph (Fig. 11a), we group the nodes according to their predicted labels. However, note that for the actual implementation, we compute the object graphs by taking the full posterior distributions and connect each superpixel node to the central unknown region. Our method produces very similar object graphs for the unknown regions, which enables them to be grouped despite their heterogeneous appearances.



(a) 2D Object-Graphs (b) 3D Object-Graphs

Fig. 11. Examples of 2D and 3D object graphs generated by our method. Our method correctly identifies the car and motorbike regions in (a) and (b), respectively, as unknowns (regions with yellow borders), and produces accurate descriptions of the surrounding familiar object-level context. Our method groups the unknown regions despite their variable appearance, due to their strong agreement in object graphs. Note that the surrounding regions that do not belong to a familiar category cannot be classified correctly (e.g., the person regions in (b)); however, the *distribution* of their known category posterior probabilities still provides meaningful appearance information that leads to accurate object-graph descriptions.

4.11 Modeling Scene Depth with 3D Object Graphs

We next evaluate the impact that the 3D object graph has on discovery. We evaluate our method on the Gould 2009 data set since it has previously been tested for computing depth estimates and is appropriate for modeling 3D scene structure from single views. The other data sets contain some images of close-up objects, which the method in [25] does not handle as well. While we choose to test on single images to demonstrate the flexibility of our approach, stereo data would also be amenable when available since their disparity maps provide depth information.

As before, we perform discovery on the regions that are deemed to be unknown. In addition, we remove any misclassified regions, i.e., true known regions misclassified as unknown, in order to isolate our analysis on the 2D versus 3D scene context description without any side effects caused by those errors. We consider in total seven neighboring regions: one region from the sky plane, one region from the ground plane, and five neighboring regions in the vertical plane; empirically, we find that the regions generated from the occlusion boundary segmentation algorithm [25] tend to correspond well with the true number of objects in the image.

Fig. 12 shows the results, compared against the 2D object-graph descriptor on the same set of unknown regions. The 3D object graph outperforms the 2D object graph. This can be attributed to the fact that the data set is mostly composed of natural scene images, where 3D geometry estimates are more reliably computed in [25]. Furthermore, the 3D object graph strictly matches regions that belong to the same geometric plane (e.g., sky regions are only compared against each other). In this way, the scene structure is retained in the comparison, providing matching scores that are more robust to camera pose variations. Nonetheless, the 2D object graph still performs quite well, which indicates that modeling spatial layout in the 2D image plane is often sufficient to provide reliable object-level context descriptions.

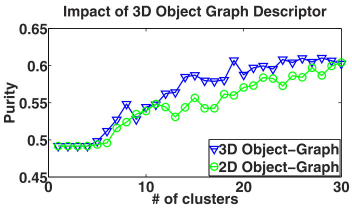
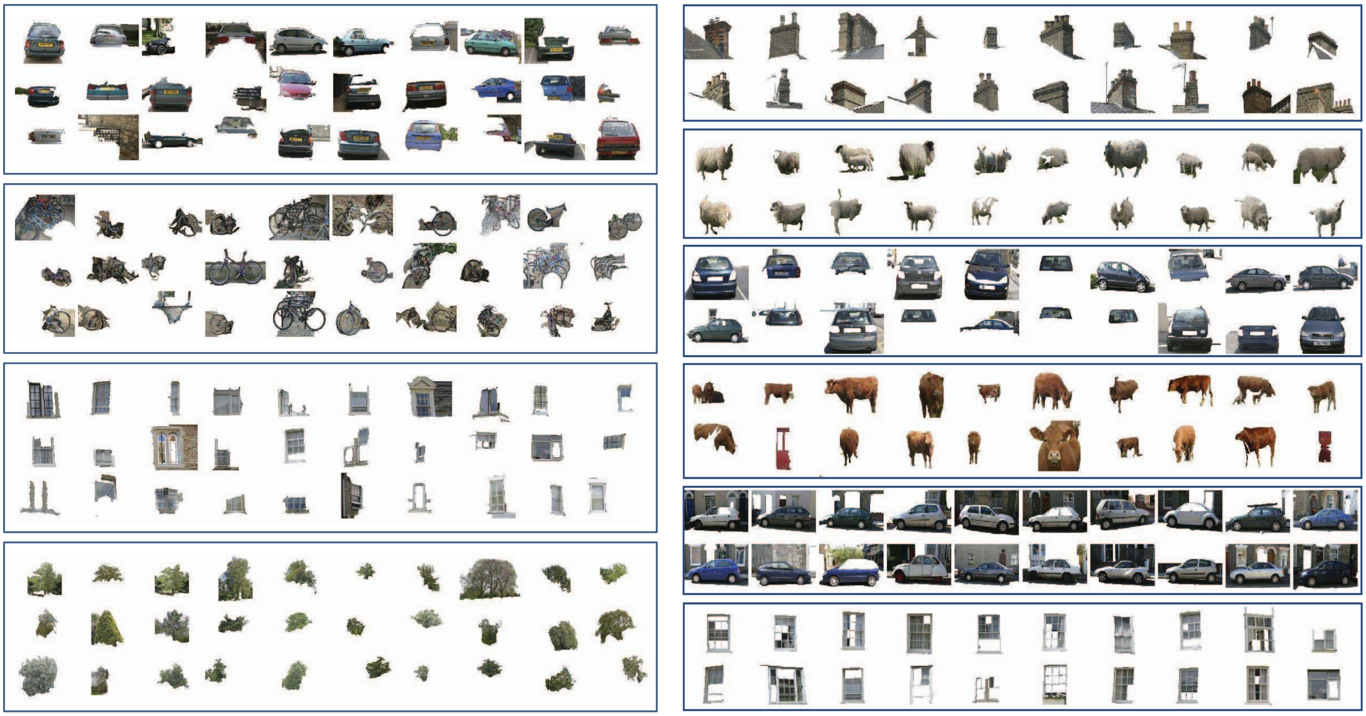


Fig. 12. Comparison of the 3D object-graph descriptor to the 2D object-graph descriptor on the Gould 2009 data set. The 3D descriptor is able to exploit the scene geometry prevalent in the data to produce more accurate descriptions of spatial context.

4.12 Discovered Categories: Qualitative Results

Finally, we provide qualitative image examples of what our method discovers. Fig. 13 shows examples of discovered categories from the 3,457 MSRC-v0 images using our 2D object-graph approach for $k = 30$. We show two sets of qualitative results using different methods to generate the candidate object regions: one using Normalized Cuts and the other using the hierarchical segmentation engine of [40]. This lets us analyze the influence that higher quality segmentations have on qualitative discovery accuracy. The cluster images are sorted by their degree (top left is highest, bottom right is lowest) as computed by the affinity matrix: $D(s_m) = \sum_{l \in L} K(s_m, s_l)$, where L denotes the cluster containing segment s_m. We show the top 20-30 regions for each cluster, removing overlapping regions and limiting to only one region per image.

The resulting groups show good semantic consistency (here, we see windows, cars, bicycles, trees, chimneys, sheep, and cows). Notably, our clusters tend to be more inclusive of intraclass appearance variation than those that could be found with methods that rely only on appearance such as [2], [1], [4], [5]. For example, note the presence of side and frontal/rear views in the sheep, car, and cow clusters (see the first row in Fig. 13a and second to fourth rows in Fig. 13b), and the distinct types of windows that get grouped together (see the third row in Fig. 13a and the last row in Fig. 13b). Our algorithm also discovers cars and buildings as a single category, which often co-occur and are segmented together (see the fifth row in Fig. 13b). This makes sense since their regions have similar appearance and similar surrounding context (i.e., road below). The segmentation quality of the discoveries made using the regions from [40] is better than those made using Normalize Cuts, which shows that better candidate object regions lead to higher quality discoveries. Overall, these results indicate that boosting affinities using both appearance and object-level context lead to semantically coherent discoveries.



(a) Discovery using NCuts regions

(b) Discovery using OWT-UCM regions

Fig. 13. Examples of discovered categories for the MSRC-v0 using (a) Normalized Cuts [28] regions and (b) Oriented WaterShed Transform—Ultrametric Contour Map [40] regions. Our clusters show good semantic consistency and tend to be more inclusive of intraclass appearance variation than those found using appearance alone. For example, note the presence of side and frontal/rear views in the car, cow, and sheep clusters, and the distinct types of windows that get grouped together. When clustering with appearance alone, it would not be possible to realize the consistency across such varying viewpoints.

5 Conclusions and Future Work

We developed an algorithm that models the interaction between familiar categories and unknown regions to discover novel categories in unlabeled images. We evaluated our approach on several benchmark data sets and showed that it leads to significant improvements in category discovery compared to strictly appearance-based baselines.

In future work, we would like to extend the system to be used in a semiautomatic loop, where an annotator labels the meaningful discovered clusters, which would then become the familiar objects for which a classifier can be trained. This would expand the object-level context for future discovery and continually increase the number of known categories.

Admittedly, known/unknown detection, or more generally "novelty detection," is a very difficult problem. We would like to investigate ways of providing more robust known/unknown decisions, either avoiding it all together by directly the known/unknown confidences into the clustering, or by using constraints and/or input from human interactions. Finally, though shown here only in the unsupervised setting, the proposed object graph may also be useful in the supervised setting, for example, for top-down image segmentation.

Acknowledgments

The authors thank Alex Sorokin for helping them with the MTurk data collection, Bryan Russell for sharing his code, and Alyosha Efros for helpful discussions, and the anonymous reviewers for their thoughtful suggestions. This research was sponsored in part by the US Army Research Laboratory under Cooperative Agreement Number W911NF-10-2-0059, DARPA CSSG, and US National Science Foundation (NSF) CAREER IIS-0747356. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, US Defense Advanced Research Projects Agency (DARPA), or the Government. The US Government is authorized to reproduce and distribute reprints for Government policies notwithstanding any copyright notation herein.

References

[1] K. Grauman and T. Darrell, "Unsupervised Learning of Categories from Sets of Partially Matching Image Features," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, 2006.

[2] B. Russell, A. Efros, J. Sivic, W. Freeman, and A. Zisserman, "Using Multiple Segmentations to Discover Objects and Their Extent in Image Collections," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, 2006.

[3] D. Liu and T. Chen, "Unsupervised Image Categorization and Object Localization Using Topic Models and Correspondences between Images," *Proc. IEEE Int'l Conf. Computer Vision*, 2007.

[4] G. Kim, C. Faloutsos, and M. Hebert, "Unsupervised Modeling of Object Categories Using Link Analysis Techniques," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[5] Y.J. Lee and K. Grauman, "Foreground Focus: Unsupervised Learning from Partially Matching Images," *Int'l J. Computer Vision*, vol. 85, pp. 143-166, 2009.

[6] A. Kaplan and G. Murphy, "The Acquisition of Category Structure in Unsupervised Learning," *Memory and Cognition*, vol. 27, pp. 699-712, 1999.

[7] R. Weischedel, "Adaptive Natural Language Processing," *Proc. Workshop Speech and Natural Language,* 1990.

[8] S. Gould, R. Fulton, and D. Koller, "Decomposing a Scene into Geometric and Semantically Consistent Regions," *Proc. IEEE Int'l Conf. Computer Vision,* 2009.

[9] Y.J. Lee and K. Grauman, "Object-Graphs for Context-Aware Category Discovery," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2010.

[10] D. Dueck and B. Frey, "Non-Metric Affinity Propagation for Unsupervised Image Categorization," *Proc. IEEE Int'l Conf. Computer Vision,* 2007.

[11] M. Cho, Y.M. Shin, and K.M. Lee, "Unsupervised Detection and Segmentation of Identical Objects," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2010.

[12] L. Fei-Fei, R. Fergus, and P. Perona, "A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories," *Proc. IEEE Int'l Conf. Computer Vision,* 2003.

[13] E. Bart and S. Ullman, "Cross-Generalization: Learning Novel Classes from a Single Example by Feature Replacement," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2005.

[14] A. Torralba, "Contextual Priming for Object Detection," *Int'l J. Computer Vision,* vol. 53, pp. 169-191, 2003.

[15] X. He, R. Zemel, and M. Carreira-Perpinan, "Multiscale Conditional Random Fields for Image Labeling," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2004.

[16] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint Appearance, Shape and Context Modeling for Multi-Class Object Recognition and Segmentation," *Proc. European Conf. Computer Vision,* 2006.

[17] G. Heitz and D. Koller, "Learning Spatial Context: Using Stuff to Find Things," *Proc. European Conf. Computer Vision,* 2008.

[18] T. Malisiewicz and A. Efros, "Beyond Categories: The Visual Memex Model for Reasoning about Object Relationships," *Proc. Neural Information Processing Systems,* 2009.

[19] A. Singhal, J. Luo, and W. Zhu, "Probabilistic Spatial Context Models for Scene Content Understanding," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2003.

[20] C. Galleguillos, A. Rabinovich, and S. Belongie, "Object Categorization Using Co-Occurrence, Location and Appearance," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[21] Z. Tu, "Auto-Context and Its Application to High-Level Vision Tasks," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[22] D. Parikh, C.L. Zitnick, and T. Chen, "From Appearance to Context-Based Recognition: Dense Labeling in Small Images," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[23] S. Lazebnik and M. Raginsky, "An Empirical Bayes Approach to Contextual Region Classification," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[24] D. Hoiem, A.A. Efros, and M. Hebert, "Putting Objects in Perspective," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2006.

[25] D. Hoiem, A.N. Stein, A.A. Efros, and M. Hebert, "Recovering Occlusion Boundaries from a Single Image," *Proc. IEEE Int'l Conf. Computer Vision,* 2007.

[26] A. Saxena, M. Sun, and A. Ng, "Make3D: Learning 3D Scene Structure from a Single Still Image," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 31, no. 5, pp. 824-840, May 2009.

[27] D. Hoiem, A.A. Efros, and M. Hebert, "Geometric Context from a Single Image," *Proc. IEEE Int'l Conf. Computer Vision,* 2005.

[28] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[29] T. Malisiewicz and A.A. Efros, "Improving Spatial Support for Objects via Multiple Segmentations," *Proc. British Machine Vision Conf.,* 2007.

[30] F. Bach, G. Lanckriet, and M. Jordan, "Multiple Kernel Learning, Conic Duality, and the SMO Algorithm," *Proc. Int'l Conf. Machine Learning,* 2004.

[31] A. Vedaldi and S. Soatto, "Relaxed Matching Kernels for Object Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[32] T. Gartner, P. Flach, and S. Wrobel, "On Graph Kernels: Hardness Results and Efficient Alternatives," *Proc. Ann. Conf. Computational Learning Theory,* 2003.

[33] H. Kashima, K. Tsuda, and A. Inokuchi, "Kernels on Graphs," *Kernels and Bioinformatics,* 2004.

[34] A. Ng, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," *Proc. Neural Information Processing Systems,* 2001.

[35] A. Bosch, A. Zisserman, and X. Munoz, "Representing Shape with a Spatial Pyramid Kernel," *Proc. Int'l Conf. Image and Video Retrieval* 2007.

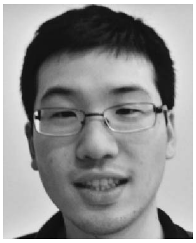
[36] J. Platt, "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," *Advances in Large Margin Classifiers,* MIT Press, 1999.

[37] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining.* Pearson Addison Wesley, 2005.

[38] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 33, no. 5, pp. 898-916, May 2011.

[39] Y.J. Lee and K. Grauman, "Collect-Cut: Segmentation with Top-Down Cues Discovered in Multi-Object Images," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2010.

[40] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "From Contours to Regions: An Empirical Evaluation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.



Yong Jae Lee received the BS degree in electrical engineering from the University of Illinois at Urbana-Champaign in 2006 and the MS degree in electrical and computer engineering from the University of Texas at Austin in 2008. He is currently working toward the PhD degree in electrical and computer engineering at the University of Texas at Austin. His research interests are in computer vision and machine learning. He is a student member of the IEEE.

Kristen Grauman received the BA degree in computer science from Boston College in 2001 and the SM and PhD degrees in computer science from the Massachusetts Institute of Technology (MIT) in 2003 and 2006, respectively, before joining the University of Texas at Austin in 2007, where she is the Clare Boothe Luce Assistant Professor in the Department of Computer Science. Her research focuses on object recognition and visual search. She has published more than 40 articles in peer-reviewed journals and conferences, and work with her colleagues received the Marr Prize at the International Conference on Computer Vision (ICCV) in 2011 and the Best Student Paper Award at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) in 2008. She is a Microsoft Research New Faculty Fellow and a recipient of the US National Science Foundation (NSF) CAREER Award and the Howes Scholar Award in Computational Science. She serves regularly on the program committees for the major computer vision conferences and is a member of the editorial board for the *International Journal of Computer Vision*. She is a member of the IEEE and the IEEE Computer Society.