Recognition by Association via Learning Per-exemplar Distances

Tomasz Malisiewicz Alexei A. Efros
The Robotics Institute, Carnegie Mellon Universtity
{tmalisie,efros}@cs.cmu.edu

Abstract

We pose the recognition problem as data association. In this setting, a novel object is explained solely in terms of a small set of exemplar objects to which it is visually similar. Inspired by the work of Frome et al., we learn separate distance functions for each exemplar; however, our distances are interpretable on an absolute scale and can be thresholded to detect the presence of an object. Our exemplars are represented as image regions and the learned distances capture the relative importance of shape, color, texture, and position features for that region. We use the distance functions to detect and segment objects in novel images by associating the bottom-up segments obtained from multiple image segmentations with the exemplar regions. We evaluate the detection and segmentation performance of our algorithm on real-world outdoor scenes from the LabelMe [15] dataset and also show some promising qualitative image parsing results.

1. Introduction

Object recognition is one of the holy grail problems in computer vision. Yet, the very notion of "recognition" is not well defined. Usually, this is assumed to mean *object naming* – given an image, the goal is to name the depicted objects (and possibly show the objects' spatial extent). But since our language does not have a name for every possible object instance, this requires that object *categories* be used for naming purposes. However, going from objects to object categories is an extremely noisy and lossy process: "a picture is worth a thousand words" – not one or two typically used for categorization. It is in fact not at all clear whether categorization, which is primarily a linguistic construct, is useful when talking about vision. For example, functional categories often exhibit visual polysemy – object instances that have visually nothing to do with each other (e.g. "chair"). Moreover, categories are language dependent – an object category in one language might not exist in another. Yet another source of visual polysemy particular to 2D image sets is view-dependence. Taken on its own, a side-view of a car has visually nothing in common with a frontal view of a car (Figure 2). Therefore, trying hard to make the same car detector fire on both seems counterproductive. Finally, object categorization is not even consistent across individuals. Consider, for instance, the LabelMe dataset [15] where human labelers can choose any English word/phrase they like for object annotation. Figure 2 (left) shows a typical example of visual synonyms – two visually similar objects that have been arbitrary assigned different labels ("building" vs. "house"). In short, object categorization is a very difficult, delicate matter. But is it absolutely nessesary for object recognition?

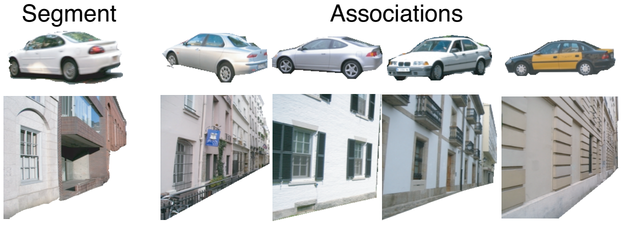

Figure 1. Recognition by Association. Two example object detections from a subset of LabelMe [15] are shown. Each detection consists of a segment extracted from an image as well as a list of object associations.

In this work, we are advocating a different way of thinking about recognition – not as object naming, but rather as object association. The idea, suggested by evidence from cognitive science, is that the central question of recognition might not be "What is it?" but rather "What is it *like*?" [2]. The etymology of the very word "re-cognize" (to know again) supports the view that association plays a key role in recognition. Under this model, when faced with a novel object, the task is to associate it with the most similar objects in one's memory. These remembered objects, in turn, provide the meta-data (e.g. object name, its context, associated actions, etc) needed to interpret the novel object.

An important benefit of object association over object naming is that there is no need to divide the world up into rigid, pre-defined categories *a priori*. Instead, each object instance uses its nearest neighbors to infer its own identity, as general or as specific as the available data allows. For example, if our dataset doesn't contain many cars, then the best that we can say about a new car instance is that it's

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

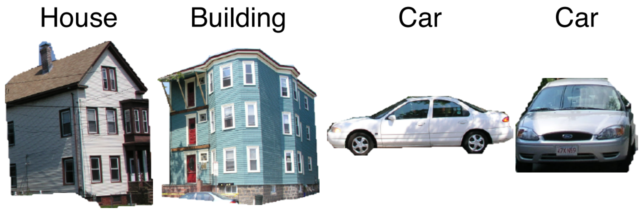

Figure 2. Typical examples of visual synonyms and visual polysemy that are common in LabelMe [15] annotations. Visual synonyms: two objects that are visually quite similar but have different class labels (left). Visual polysemy: two objects that have nothing in common *visually* but are labelled to be the same class (right).

a "car". But as the number of different cars in the dataset grows, we should be able to find very specific car matches which will allow us to recognize the same object instance as "red Honda Accord".

Of course, despite the benefits, posing object recognition as data association is not an easy task. One requirement is a very large dataset, rich enough to contain many different objects and many instances of each. Recently, with the appearance of such large image collections, several systems have shown that simple k-nearest-neighbor (kNN) approaches can often perform surprisingly well [21, 9, 14]. However, all these methods match the image as a whole, which effectively limits them to operating on the coarse scene level (there is simply not enough data in the world to observe all possible objects in all possible configurations). To match individual objects within scenes, we must partition the image into chunks which are small enough to be matchable in a reasonably-sized database, but large enough to encode specific objects, not generic "visual words". This requires addressing the difficult image segmentation problem head on.

Worse yet, objects can exhibit similarity on many different, often contradictory, levels: shape, size, color, texture, etc. For example, Adelson divides the world into "things" (such as cars, people) and "stuff" (grass, pavement, ice cream, etc.) [1]. For "things", like cars, object shape is an important cue whereas object color is usually not. But for "stuff", like grass, which doesn't own its boundaries, shape is useless (in fact, detrimental) but color and texture are extremely important. Therefore, to find what a given object instance is similar to, it is imperative that the right distance metric for that instance be used. But, of course, to know the right distance metric requires knowing what that object is! As is often the case in vision, we are faced with a difficult chicken and egg problem.

In this paper, we take the first steps in addressing the issues outlined above, toward the ultimate goal of real-world image understanding. We propose a segment-centric, exemplar-based system for establishing object association within a large, inconsistently labeled image collection. The main contributions of our work are:

- Posing the recognition problem as **data association**. In this setting, a novel object is defined/explained solely in terms of a small set of exemplars to which it is similar. At the recognition stage, there is no mention of labels, categories or classes. This data-driven definition requires better ways of object matching, leading to the following algorithmic contributions:

- Improving nearest neighbor performance by **learning interpretable per-exemplar distances**. Inspired by the work of Frome et al. [7, 8], we learn individual distance metrics for each of our exemplars. But unlike theirs, our distances are interpretable on an absolute scale, and can be thresholded to perform detection. In addition to learning a distance, we also determine for each exemplar the subset of other exemplars that are similar to it. This allows us to capture visual relationships within our dataset that were not reflected in the labels.

- Partitioning the input image into parts small enough for object matching via **recognition-based object segmentation**. First, a large number of proposal segments is generated using the multiple segmentation framework [10, 16, 12]. Then, data-driven association is used to find segments that are more likely to represent objects.

The rest of this paper is organized as follows. In Section 2 we review previous work in the area. In Section 3 we discuss our algorithm for learning associations between objects. In Section 4 we present an algorithm for finding data associations in novel images. Finally, we conclude with a discussion of results and future directions in Section 5.

2. Background

Most object recognition work can be divided into two camps: object detection and object classification. Object detection systems are concerned with localizing objects in images but focus on a single object category at a time while treating the rest of the image as background clutter. State-of-the-art detectors exist for only a few object categories: cars [17], pedestrians [5], and faces [17, 22] (all compact objects, well modeled by a sliding rectangle). Methods that interleave segmentation and detection have also been proposed (e.g. [11]). Object classification systems, on the other hand, have been recently developed that handle a large number of object categories (e.g. the Caltech 101 dataset [6, 24, 8]). However, such multi-class object classification avoids the localization problem by only dealing with images that contain a single object. While making progress dealing with a large number of categories is very important, such contrived images are not very representative of what the real world looks like. A real-world scene depicts the interplay between many different types of objects making localization a very important part of successful scene understanding.

Recently, several groups have been working on approaches for localizing object instances coming from a large number of categories. On one hand, bottom-up MRF-based methods (e.g. [19]) assign per-pixel or per-patch object category labels based on local texture appearance and global label propagation. However, such models are too texture-oriented and do not elegantly handle multiple object instances [23]. On the other hand, global approaches [20, 14] utilize information over the entire image (the gist of the scene) to provide guidance for many separate object detectors. While considering the global scene information is indeed very important, at the end these methods are still limited by the low-level object detectors that perform well only for small number of rectangular-looking objects.

Unlike the bottom-up (pixel-based) or top-down (gist-based) approaches described above, in this work we argue for a mid-level, segment-centric view. We define a segment as a contiguous region extracted from an image. We use segments both in training and at run-time to detect and segment objects. Our approach is exemplar-based, which detects objects by associating segments in an input image with exemplars from the dataset. We envision our method to be the front-end for a complete image parsing system, therefore it's important that we perform well with respect to both multi-class recognition and object segmentation.

Our method is most similar to the object detection system of Chum et al. [3] as well as the classification system of Frome et al. [7, 8]. In the work of Chum et al. (the recent PASCAL challenge winner), categories are represented via exemplars, but the underlying assumption is that all exemplars from the same class are similar. Thus, Chum et al. require class-wise *and* aspect-wise labeling of training data. Unfortunately this type of labeling is tedious and difficult to obtain for datasets of significant size. Another major limitation of this approach is that it was only shown to work well for compact, rectangle-shaped objects. It is not clear how such an approach will work on the multitude of common objects whose shape is not well approximated by a rectangle.

The work of Frome et al. [7, 8] deals with a large number of object categories and learns how to compare exemplars via local distance functions for the task of image classification (on Caltech 101). While the distances being learned are on the same scale and can be compared to each other, they are not meaningful in absolute terms. This is not a problem for a forced classification task (Caltech 101) where the most likely class is assigned to each input. However, this approach is not suitable for recognition tasks where many input patterns should not correspond to any objects and should be given very large absolute distances. Additionally, their distance functions are applied to entire images – essentially bypassing the problems inherent in segmentation and localization.

3. Learning Object Similarity

Rather than building models that measure the similarity between classes we only want to quantify the degree of similarity between an input and an exemplar. Of course, similarity is defined differently for different types of objects in the world. We tackle this problem by learning a separate combination of elementary distances (such as color, texture, shape, etc) for *each exemplar in our database*. To make things difficult, typical human object labels are not good enough to make the assumption that an exemplar should be similar to all other exemplars with the same label (see Figure 2). We instead propose a largely data-driven approach which weakly uses the object labels to automatically learn for each exemplar e a distance function *and* which subset of exemplars are similar to e.

Focusing on the object association paradigm, we train our distance functions to return interpretable distances which can be reasoned about in absolute terms. We say that an input associates with exemplar e if e's distance function returns a distance less than 1. As opposed to kNN methods, each potential input can associate with a variable number of exemplars – common objects should associate with many exemplars, rare objects might only associate with a single exemplar, and bad inputs (or simply never seen before inputs) shouldn't have any associations.

3.1. Dataset

We are ultimately interested in parsing a scene into its constituent objects – understanding as much as we can about an input image. Doing so for a reasonably general class of images requires handling a large number of different objects that occur in everyday life. Therefore, the choice of the right training data is of the utmost importance. Of all the currently available datasets, the only one containing a large number of real-world scenes, with a wide variety of everyday objects that are not only labeled but also segmented, is the LabelMe dataset [15]. LabelMe is an ongoing online image annotating collaboration involving many labelers. As a result, not only are the images user-contributed, spanning a wide range of scenes, but users are free to label each object with any English text string they like, providing a good sampling of the distribution of object names "in the wild".

We use a subset of LabelMe which consists of over 5000 images. After ignoring tiny objects, we clean up the object annotations by discarding auxiliary words from the labels (using the function provided in LabelMe toolkit), and keep all objects whose unique label occurs at least 5 times. This gives us a total of $12,905$ objects spanning 171 unique labels. Given the ambiguity of the user-defined "wiki-labels", we don't want to say that the world is made up of a fixed number of object classes defined by these labels, choosing instead the object association paradigm.

Type	Name	Dimension
Shape	Centered Mask	32x32=1024
	BB Extent	2
	Pixel Area	1
Texture	Right Boundary Tex-Hist	100
	Top Boundary Tex-Hist	100
	Left Boundary Tex-Hist	100
	Bottom Boundary Tex-Hist	100
	Interior Tex-Hist	100
Color	Mean Color	3
	Color std	3
	Color Histogram	33
Location	Absolute Mask	8x8=64
	Top Height	1
	Bot Height	1

Table 1. The 14 Region-Based Features used to represent objects. Elementary distances are simply the L_2 distances between corresponding feature vectors.

3.2. Segment Features

In our work, we represent exemplars as segments. Each object segment is characterized with $N_F = 14$ different features. Elementary distances are defined for each of these features to be simply the L_2 norm between the feature representations. The features roughly capture different aspects of shape, texture, color, and image location for an image segment (see Table 1). To capture information about shape we compute: the centered object mask in a canonical 32×32 frame, the size of the region, and the size of region's bounding box. To capture texture we compute normalized texton histograms in the interior of the object, and, separately, along the boundaries of the object. For color we compute the mean RGB-value, its standard deviation, as well as a color histogram. Finally, to capture knowledge about the position of the segment in an image, we compute a coarse (blurred) 8×8 absolute segmentation mask as well as the height of the top-most and bottom-most pixel in the region.

3.3. Learning Distance Functions

Our distance functions are positive linear combinations of elementary distances. Each exemplar has its own distance function – we denote exemplar e's distance function as D_e. We define $\mathbf{d}_{ez}$ [1] to be the $N_F + 1$ dimensional positive "distance" vector between e and input z (the j-th component of $\mathbf{d}_{ez}$ is just the L_2 distance between the $j - th$ feature vectors of e and z). Each distance function is parametrized by weight vector $\mathbf{w}_e$ and takes the form:

$$D_e(z) = \mathbf{w}_e \cdot \mathbf{d}_{ez} \tag{1}$$

In addition to $\mathbf{w}_e$, each exemplar is associated with a binary vector $\boldsymbol{\alpha}_e$. The length of $\boldsymbol{\alpha}_e$ is equal to the number of exemplars with the same label as e. The non-zero elements of $\boldsymbol{\alpha}_e$ are precisely the exemplars that should be similar to e. We learn $\mathbf{w}_e$ and $\boldsymbol{\alpha}_e$ simultaneously while keeping each exemplar's learning problem *independent* of the other distance functions. The learning problem is formulated as follows (we drop the e subscript for clarity):

$$\{\mathbf{w}^*, \boldsymbol{\alpha}^*\} = \operatorname*{argmin}_{\mathbf{w},\boldsymbol{\alpha}} f(\mathbf{w}, \boldsymbol{\alpha}) \tag{2}$$

$$f(\mathbf{w}, \boldsymbol{\alpha}) = \sum_{i \in C} \alpha_i L(-\mathbf{w} \cdot \mathbf{d}_i) + \sum_{i \notin C} L(\mathbf{w} \cdot \mathbf{d}_i) \tag{3}$$

subject to the constraints that $\mathbf{w} \geq 0$, $\alpha_j \in \{0, 1\}$, and $\sum_j \alpha_j = K$ (the minimum number of exemplars we force to be similar to e). $L(\cdot)$ is any positive loss function, and C_e is the set of all exemplars with the same label as e. Without the $\boldsymbol{\alpha}$ parameter and with no constraint on $\mathbf{w}$, this is just the primal form that many convex statistical learning techniques (such as Logistic Regression and SVMs) can be cast in. In our case, the positivity of $\mathbf{w}$ is meant to ensure that a large elementary distance can never imply a higher degree of similarity.

Since the presence of the binary $\boldsymbol{\alpha}$'s renders the problem non-convex, we proceed iteratively estimating $\boldsymbol{\alpha}$ given $\mathbf{w}$ and estimating $\mathbf{w}$ given $\boldsymbol{\alpha}$. During each iteration, we are guaranteed to never increase the value of our objective function (Equation 3) and thus efficiently find a local minimum. We start with an initial distance function $\mathbf{w}^0$ and proceed as follows:

$$\boldsymbol{\alpha}^k = \operatorname*{argmin}_{\boldsymbol{\alpha}} \sum_{i \in C} \alpha_i L(-\mathbf{w}^{\mathbf{k}} \cdot \mathbf{d}_{\mathbf{i}}) \tag{4}$$

$$\mathbf{w}^{k+1} = \operatorname*{argmin}_{\mathbf{w}} \sum_{i : \alpha_i^k = 1} L(-\mathbf{w} \cdot \mathbf{d}_i) + \sum_{i \notin C} L(\mathbf{w} \cdot \mathbf{d}_i) \tag{5}$$

Given $\mathbf{w}^k$, we minimize equation 4 by setting α_i equal to 1 for the K smallest values of $L(-\mathbf{w} \cdot \mathbf{d}_{\mathbf{i}})$, and 0 elsewhere. Given $\boldsymbol{\alpha}^k$ – which essentially selects which exemplars should influence the decision boundary – the problem of solving equation 5 is just the classical convex statistical learning problem. This procedure converges when $\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k$.

In particular, we use the squared hinge-loss function and solve the Support Vector Machine problem in the primal. For every exemplar we initialize the distance function with $\mathbf{w}^0 = D_{texton}$ which only uses the L_2 distance between interior texton histograms. From all combinations of heuristically defined distances we experimented with, D_{texton} did the best for a wide array of object types. We set $K = 10$ for all exemplars. After solving each optimization problem, we scale the resulting distance function such that a distance value of 1 (instead of 0) corresponds to the decision boundary and a value of 0 (instead of $-\mathbf{w}_{N_F+1}$) corresponds to perfect similarity.

Similarly to [7], we focus on linear decision boundaries and pose each distance function learning problem as a SVM

[1] To handle a bias term, we concatenate a fixed -1 to the vector of elementary distances. $\mathbf{d}_{ez} = [\mathbf{d'}_{ez}; -1]$

stop sign	sign	7.8%	road highway	road	3.4%
pole	streetlight	6.7%	painting	picture	3.4%
motorcycle	motorbike	6.2%	sidewalk	road	3.2%
mountains	mountain	6.2%	cloud	sky	3.1%
ground grass	sidewalk	3.7%	grass	ground grass	3.1%
grass	lawn	3.6%	mountain	mountains	2.7%

Table 2. Top dozen label confusions discovered after distance function learning. For example, 7.8% of the time a "stop sign" wanted to associate with a "sign."

convex optimization problem. While we learn the distance functions independently, the inclusion of a bias term in our problem results in interpretable distances without any post-processing (unlike [7]).

After learning the distance functions, we apply each exemplar's distance function to all of the other exemplars and consider the support set $Supp(D_e)$ as $z \in Supp(D_e) \leftrightarrow D_e(z) < 1$. In practice the resulting support sets wildly vary in size. For exemplars from generic classes such as "sky" where we expect many skies to be rather similar, $|Supp(D_e)| >> K$. The support set can also be very small – which happens when its corresponding exemplar is either not visually distinctive or ambiguously/incorrectly labeled. We prune away the exemplars with an empty support set. Several learned distance functions and the top elements in their support sets are shown and compared to the neighbors given a simple texton-histogram distance in Figure 3. The learned distance functions are doing a good job at combining elementary distances to measure similarity. Notice that an exemplar's support doesn't always contain exemplars with the same label. In particular an exemplar with the label "standing person woman" was deemed similar to the target exemplar with label "person" even though they are distinct labels. We measure how often this happens and show the top few elements of the label confusion matrix in Table 2. Notice that most of these confusions correspond to visual synonyms.

In order to determine if the distance functions are overfitting, we consider a segment-labeling task utilizing over 1000 objects extracted from a held-out subset of LabelMe. In the segment-labeling task, we are given a ground truth segmentation mask and choose the label from the closest exemplar. We consider a set of distance thresholds and compute the precision versus recall curve. Precision measures the probability that a returned label is identical to the ground truth label and in our case recall measures the fraction of segments that get labeled. As a baseline, we compare the performance of our learned distance functions to a nearest-neighbor classifier using a texton histogram distance. The precision-recall curve can be seen in Figure 4. If we only interpret the distance functions that return a value below 1.0 (what our learning formulation suggests is the best thing to do), we obtain labels for 60% of objects that are correct 91% of the time. This suggests that the learned distance functions are providing a very meaningful distance for recognition. In addition, the overall high precision for both distances in the segment-labeling task supports the observation that correct spatial support for objects makes recognition significantly easier [12].

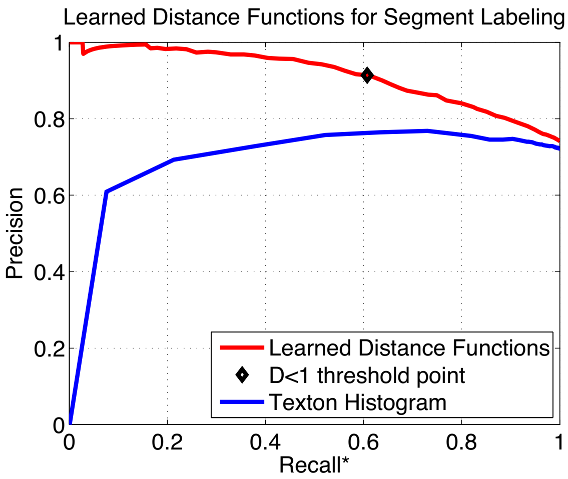

Figure 4. Ground-truth segment labeling using Per-Exemplar Distance Functions and Texton Histograms. A different distance function is learned for each exemplar. Note that the Distance Functions yield high precision when $D < 1$ suggesting that the returned distance is a good measure of recognition confidence.

4. Object Segmentation via Recognition

We already saw that the learned distance function can be used to determine the identity of a ground-truth segment, but how can we use them to segment objects inside novel, unlabeled images? We tackle this problem in two steps. We use the multiple segmentation approach [16, 12] – which was shown to provide good spatial support for many different object types – to generate a large collection of candidate segments. We then keep the set of segments that associate with some of our exemplars.

4.1. Multiple Segmentations

We use a variant of the multiple segmentation approach [16, 12], to generate a "soup of segments" in a purely bottom-up fashion. In particular, we vary the parameters of two segmentation engines – Mean-Shift based EDISON [4] and Normalized Cuts [18] – to generate multiple image segmentations for every input image. It was recently shown [12] that some composite objects are very unlikely to come out as a single segment in any segmentation, but can be well approximated by a merge of a few adjacent segments. Therefore, we augment our initial soup of segments by considering the merges of 2 or 3 adjacent segments as discussed in [12]. The resulting buttom-up segment representation can provide regions with good spatial support for both shape-free "stuff" objects such as grass, road, and sky as well as fixed-extent "things" such as cars, bicycles, and people. An additional advantage of using a bottom-up mechanism to generate candidate regions is that it is independent of the number of object categories.

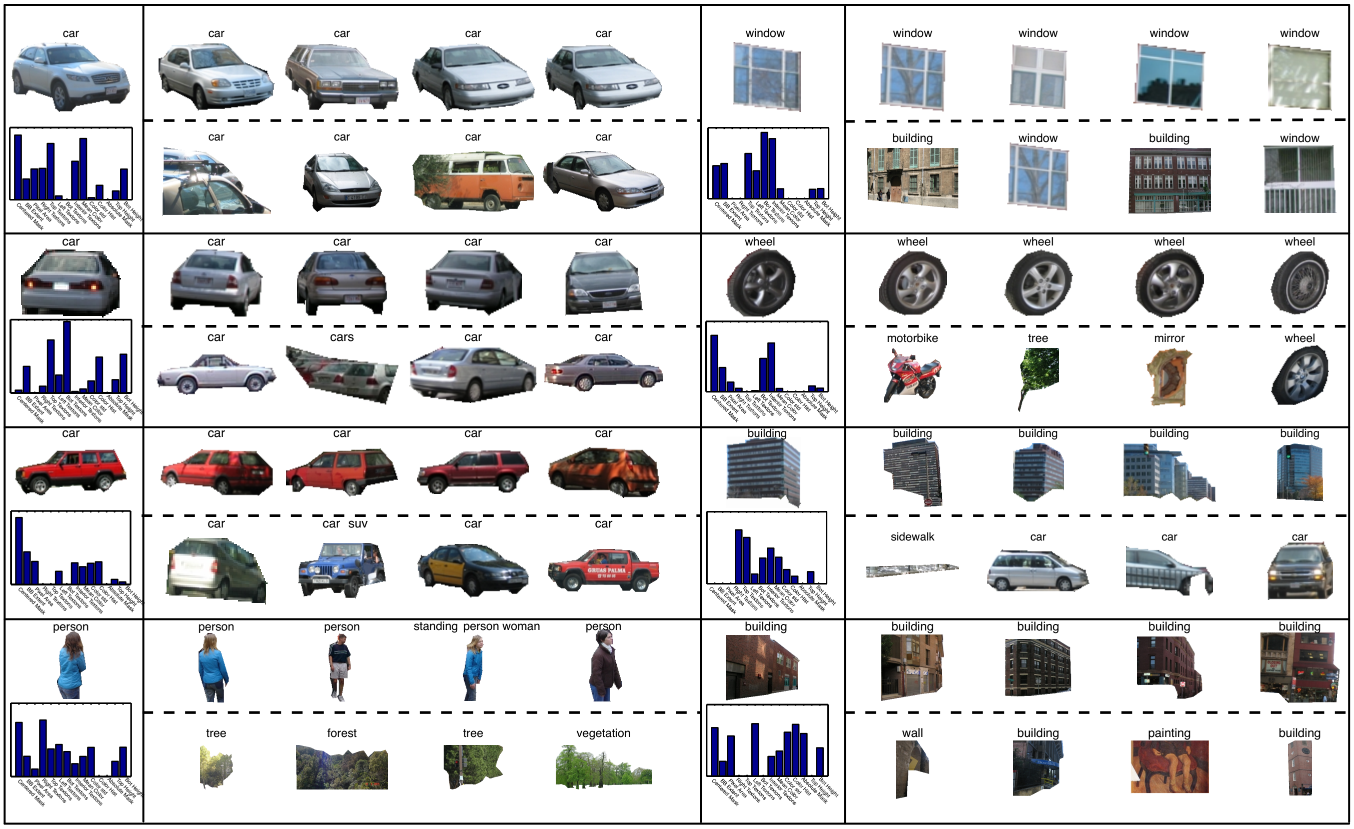

Figure 3. Data Association in the Training Set. Given an exemplar on the top left, the remaining row shows the top 4 most similar objects after learning a distance function. The distance function is visualized as a distribution over elementary distances and shown in the bottom left. The 4 exemplars on the bottom right are the 4 most similar objects with respect to the texton histogram distance.

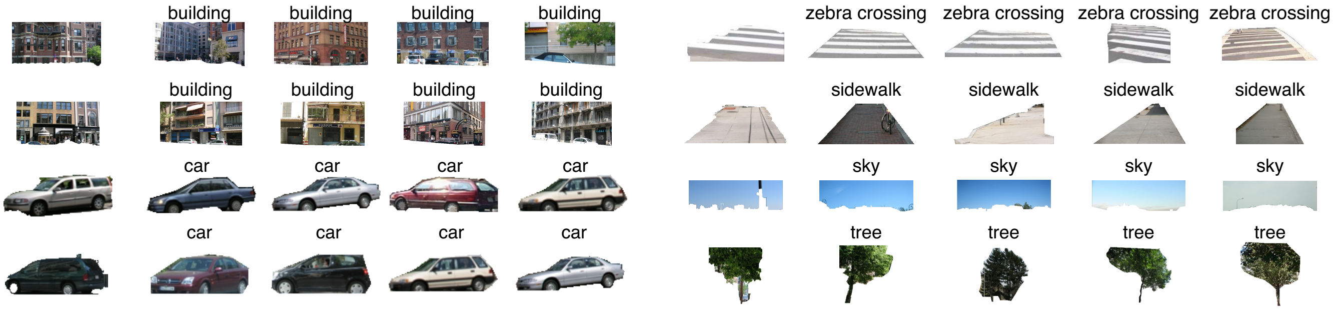

Figure 5. Data Association in the test set. Each example shows a bottom up segment and its top 4 exemplar associations.

4.2. Recognizing Good Segments

The distance functions learned so far are not very good at recognizing bad segments – they never saw any in training! We thus augment the data used in distance function learning to contain a large number (over $30,000$) of bad segments which capture the appearance of patterns that do not correspond to any objects.

After we generate the soup of bottom-up segments, we compute the full matrix of distances between all exemplars and segments. We only consider the distances below 1.0 and the resulting associations are very sparse. On average, less than $.2\%$ of the potential associations are active. Qualitative examples of data association in the soup of bottom-up segments can be seen in Figure 5. Quite often, a single segment will associate with many exemplars and we construct a recognition score out of the list of associating distances. Letting E be the list of exemplars associating with segment S the recognition confidence $s(S, E)$ is constructed as follows:

$$s(S, E) = 1/\sum_{e \in E} \frac{1}{D_e(S)} \tag{6}$$

4.3. Quantitative Evaluation

Our evaluations use a test set of 147 outdoor images all coming from one specific subfolder in LabelMe (to minimize the chances of similar data being used for training and testing). This testing subset contains a total of $1,146$ objects. We quantify how well we can detect and segment objects in this test set.

For evaluation purposes, we label each object hypothesis with the most frequently occurring label among its associations. We also retain all segments that associate with at least one exemplar, and thus have multiple (potentially all correct) overlapping object hypotheses. Since we don't want to penalize for these alternative associations we define detection precision as follows: we consider an object hypothesis to be correct if it has a segment overlap score (defined as in [12]) of at least 0.5 with a ground truth region that has the same identical label as the hypothesis. We consider all objects in tandem and do not penalize for multiple correct overlapping associations. We vary the recognition confidence to create the precision versus recall curve in Figure 6.

In order to quantify how well we segment objects, for each correct detection we measure the overlap score between the associated ground truth regions and the object hypotheses. We show the average overlap score as we vary the recognition confidence and compare that to the average overlap score of the best segment in our soup of segments. The corresponding plot can be seen on the right side of Figure 6.

4.4. Toward Image Parsing

With image parsing as our ultimate goal, we believe we made considerable progress in our current work. The ability to return a small number of object hypotheses with high quality segmentation masks is crucial for image understanding. Even though the interplay between objects (e.g. [13]) is certainly a crucial component for determining the identity of all the scene's visual elements, we can still create meaningful (partial) parses using our local distance functions alone. We create an image parse from our overlapping object hypotheses as follows: given a list of object hypotheses in a single image sorted by their recognition confidence and an initially empty list of objects in the parse, we greedily place the current best object hypotheses into the list of objects in the parse while removing all hypotheses that overlap with a score of 0.5 or more. Two resulting image parsing examples can be seen in Figure 7.

5. Conclusion

We have presented an exemplar-based system which performs detection and segmentation for a large number of different objects. Based on the principle of data association, we associate a segment extracted from a novel image with visually similar exemplar(s). We have shown that the integral component of such exemplar-based systems is the learning of exemplar-specific distance functions. While more work is needed to combine the resulting object hypotheses in a meaningful way, we believe that obtaining such mid-level segment/exemplar associations is the right step in the direction of image understanding.

Acknowledgements. This research was in part funded by NSF CAREER award IIS-0546547, NSF Graduate Research Fellowship, as well as generous gift from Google.

References

[1] E. Adelson. On seeing stuff: the perception of materials by humans and machines. In *Proc. SPIE*, 2001.

[2] M. Bar. The proactive brain: using analogies and associations to generate predictions. *Trends in Cognitive Sciences*, 11(7), 2007.

[3] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *CVPR*, 2007.

[4] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, 2002.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005.

[6] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE CVPR Workshop of Generative Model Based Vision*, 2004.

[7] A. Frome and J. Malik. Image retrieval and recognition using local distance functions. In *NIPS*, 2006.

[8] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *ICCV*, 2007.

[9] J. Hays and A. A. Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (SIGGRAPH 2007)*, 26(3), 2007.

[10] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *ICCV*, Oct. 2005.

[11] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Workshop on Statistical Learning in Computer Vision, ECCV*, 2004.

[12] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, September 2007.

[13] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *Proc. ICCV*, 2007.

[14] B. Russell, A. Torralba, C. Liu, R. Fergus, and W. T. Freeman. Object recognition by scene alignment. In *NIPS*, 2007.

[15] B. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. In *IJCV*, 2007.

[16] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *Proc. CVPR*, 2006.

[17] H. Schneiderman and T. Kanade. Object detection using the statistics of parts. *IJCV*, 2002.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. PAMI*, 22(8), August 2000.

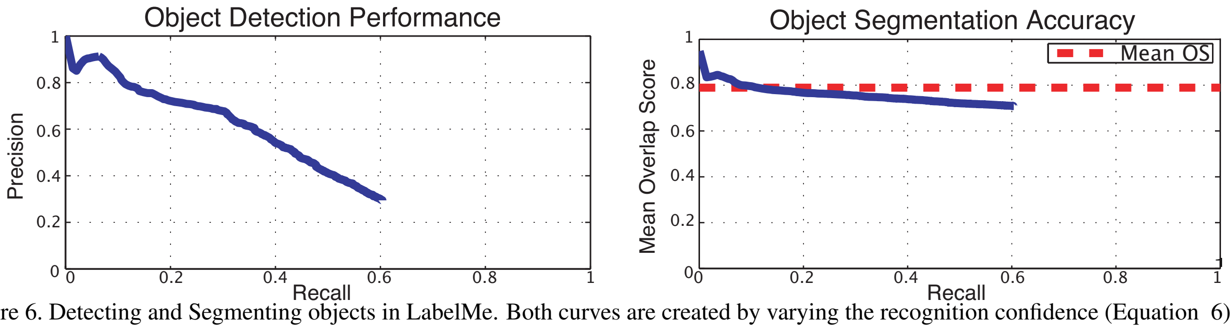

Figure 6. Detecting and Segmenting objects in LabelMe. Both curves are created by varying the recognition confidence (Equation 6). The first plot shows the precision-recall curve for the task of object detection. A detection is deemed correct if it returns the same label as well as has an overlap score (OS) greater than .5 with a ground-truth segment. The second plot shows the average segmentation quality of correct detections and compares that to the mean best overlap score of the input multiple segmentations.

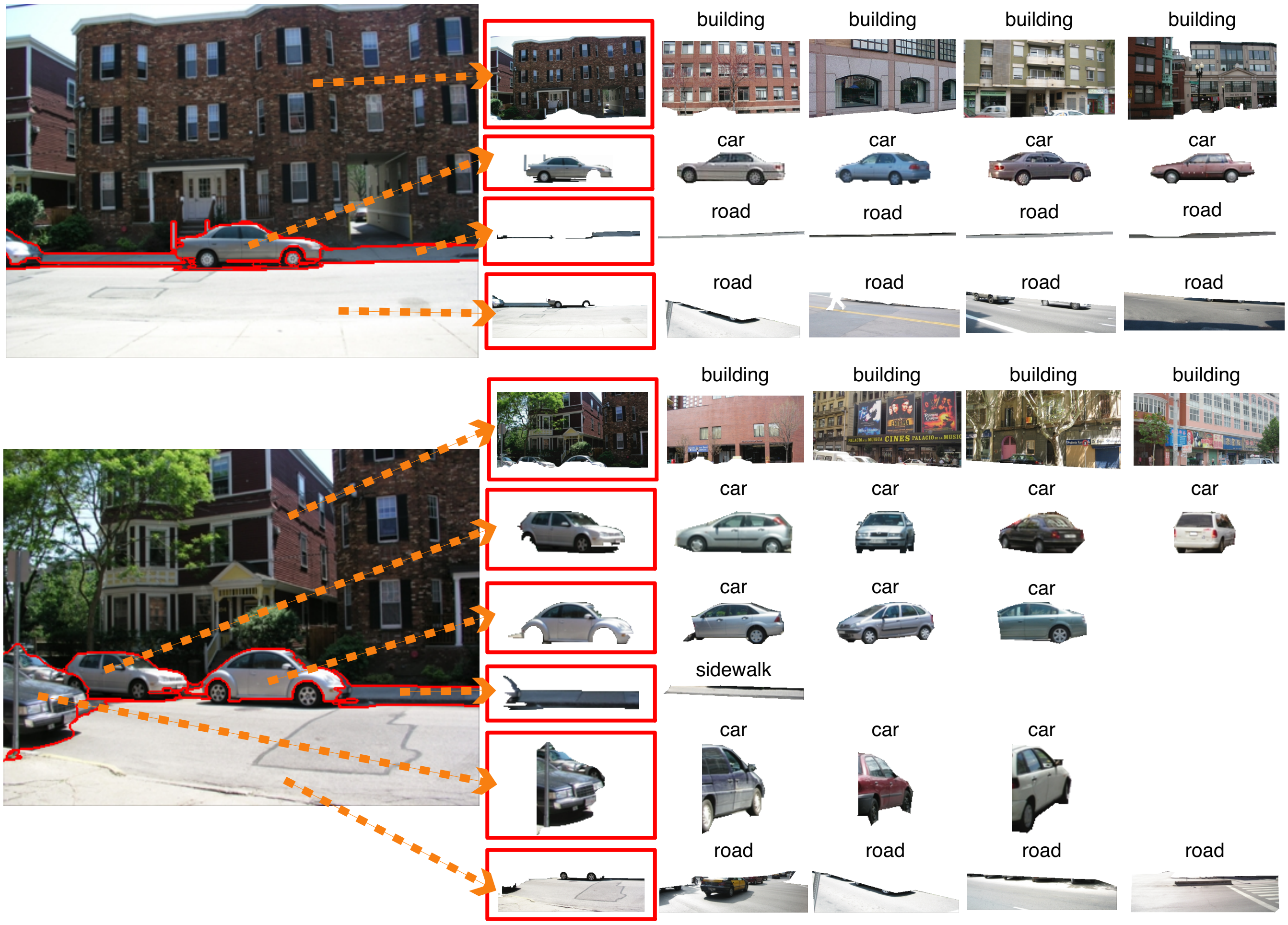

Figure 7. Parsing an image via data association. A parse is created by greedily stacking object associations that have small overlap.

[19] J. Shotton, J. M. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, 2006.

[20] A. Torralba. Contextual priming for object detection. *Int. Journal of Computer Vision*, 53(2):169–191, 2003.

[21] A. Torralba, R. Fergus, and W. T. Freeman. Tiny images. Technical Report MIT-CSAIL-TR-2007-024, MIT CSAIL, 2007.

[22] P. Viola and M. Jones. Robust real-time object detection. *2nd Intl. Workshop on Statistical and Computational Theories of Vision*, 2001.

[23] J. Winn and J. Shotton. The layount consistent random field for recognizing and segmenting partially occluded objects. In *CVPR*, 2006.

[24] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *CVPR (2)*, pages 2126–2136, 2006.