In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2006.

Unsupervised Learning of Categories from Sets of Partially Matching Image Features

Kristen Grauman and Trevor Darrell
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139
{kgrauman,trevor}@csail.mit.edu

Abstract

We present a method to automatically learn object categories from unlabeled images. Each image is represented by an unordered set of local features, and all sets are embedded into a space where they cluster according to their partial-match feature correspondences. After efficiently computing the pairwise affinities between the input images in this space, a spectral clustering technique is used to recover the primary groupings among the images. We introduce an efficient means of refining these groupings according to intra-cluster statistics over the subsets of features selected by the partial matches between the images, and based on an optional, variable amount of user supervision. We compute the consistent subsets of feature correspondences within a grouping to infer category feature masks. The output of the algorithm is a partition of the data into a set of learned categories, and a set of classifiers trained from these ranked partitions that can recognize the categories in novel images.

1. Introduction

Current approaches to object and scene recognition typically require some amount of supervision, whether it is in the form of class labels for training examples, foreground-background segmentations, or even a detailed labeling of objects' component parts. In part due to the significant expense of providing these manual annotations, such approaches are in practice restricted to relatively small numbers of classes and/or few training examples per class. Additionally, human supervision may result in unintentional biases that can be detrimental to generalization performance. An unsupervised (or semi-supervised) technique that is able to recover salient categories directly from images would relieve these burdens and possibly offer new insights into image representation choices.

In this work, we propose an efficient method to automatically learn groupings over sets of unordered local features by embedding the sets into a space where they cluster according to their partial-match correspondences. Each image is decomposed into a set of local feature descriptors. Then every set is treated as a node in a graph, where an edge between two nodes (sets) is weighted according to how well some subset of the two sets' features may be put into correspondence, with correspondence quality determined by descriptor similarity. A spectral clustering algorithm is then applied to the graph's affinity matrix to produce an initial set of image groupings. In an (optional) semi-supervised paradigm, we allow the user to select pairwise constraints between some number of input images, where constraints are in the form of "must-group" or "cannot-group" specifications. The affinity matrix is then modified to incorporate the user-supplied groupings prior to the spectral clustering step.

Spectral clustering on approximate partial-match similarity scores is efficient and produces clusters that coarsely group distinct object classes. To improve specificity, and to develop a predictive classifier that can label unseen images, we develop a method to find *prototypical* examples in each cluster that are more likely to be class inliers, and then use these prototypes to train a predictive model.

We detect prototype examples by examining the pattern of partial match correspondences within a cluster. Outlier cluster members are identified as those images that cause most images within the cluster to contribute an inconsistent subset of features in a partial match. With the assumption that outlier images will be less likely to match the same features as the majority of inlier images, we re-weight intra-cluster matching scores under a per-image mask representing the image elements that were most likely to be in correspondence when matched to other examples in the cluster.

Implied in the motivation for unsupervised learning of categories is the idea that while labeled data is expensive

and must be used frugally, unlabeled data is generally inexpensive to obtain in large quantities. Thus a critical criterion for a method intended to learn from large amounts of unlabeled data is computational efficiency; with this important consideration, we have designed a method that will scale well with both the amount of input data as well as the size of the inputs themselves.

Possible applications of the proposed method include learning object class models from unlabeled data, shot matching or scene grouping from video sequences, and content-based refinement of keyword-based image retrievals. In this paper we demonstrate the applicability to learning object categories to allow unsupervised training of discriminative classifiers.

2. Related Work

Much recent work has shown that sets of local image features are a powerful representation for recognition and retrieval (e.g., [1, 7, 8, 2]). Whereas global vector-based representations are known to be sensitive to real-world image variations, local features are often more reliably detected and matched across different examples of an object or scene under varying viewpoints, poses, or lighting conditions. It is unclear, however, how to appropriately apply conventional unsupervised learning techniques in this domain, where every example is a set of unordered feature vectors, and each set may vary in size.

Existing approaches to this problem use vector quantization to build a codebook of feature descriptors, and then transform each set input to a single vector counting the number of occurrences of each prototype feature. Conventional clustering methods or latent semantic analysis (LSA) may then be directly applied [10, 12, 3], and have been shown to yield promising results when learning object or scene categories or filtering keyword-based image retrieval outputs.

However, such approaches do not explicitly allow for "clutter" features caused by image backgrounds or occlusions, and the need to pre-compute a codebook raises computational complexity and data availability issues. In addition, it is not clear how existing techniques could accommodate the addition of small amounts of labeled data or *a priori* knowledge about pairwise constraints between particular unlabeled examples.

In general methods to solve for explicit correspondence are computationally expensive, requiring cubic time to form globally optimal assignments, and even more for assignments including higher order constraints between features. A number of approximate methods have been defined, which offer improved performance under certain restrictions.

The authors of [2] introduced a powerful recognition algorithm that uses linear programming to solve for approximate correspondences, and they showed how to use the correspondence-based metric to find regions of common spatial support for objects in labeled training examples, thus avoiding the need for manually segmented images. In [4], we introduced a kernel providing an efficient approximation of the optimal partial matching between two sets of features for discriminative classification. Sets of local image descriptors are compared in terms of how well some subset of their features may be put into correspondence. However, in the recognition frameworks of both [2] and [4], it is assumed that class labels are provided for all training images.

3. Approach

Given a collection of unlabeled images, our method produces a partition of the data into a set of learned categories, as well as a set of classifiers trained from these ranked partitions which can recognize the categories in novel images.

Each image is represented by an unordered set of local features. First, pairwise affinities reflecting partial-match feature correspondences are computed for all input images. A variable amount of supervised labels (pairing constraints) are optionally collected from the user, and the affinity matrix is adjusted accordingly. Spectral clustering is then used to recover the initial dominant clusters. Then, this clustering is distilled to sets of prototypical examples from each category by evaluating the typical "feature masks" contributing to each within-cluster partial matching. The top-ranked prototypical examples from each of the refined groupings compose the learned categories, which are used to train a set of predictive classifiers for labeling unseen examples.

3.1. Grouping Feature Sets with Partial Correspondences

Every input image is decomposed into some number of local appearance features, where each feature is a vector descriptor for the local region or patch. So given an unlabeled data set $\mathbf{U} = \{\mathbf{I}_1, \ldots, \mathbf{I}_N\}$ containing N images, image $\mathbf{I}_i$ is represented by a set $\mathbf{X}_i = \{\mathbf{f}_1, \ldots, \mathbf{f}_{m_i}\}$, where $\mathbf{f}_j$ is a descriptor vector, and m_i may vary across $\mathbf{U}$ depending on the number of features detected in each image. In our implementation we chose to use the SIFT descriptor [7], but other options are certainly possible.

The initial image groupings are formed by embedding the feature sets into a space where they cluster according to their partial-match correspondences. We use the pyramid match kernel, which we developed for discriminative classification in [4], to efficiently obtain these matchings. The pyramid match kernel computes a weighted intersection over multi-resolution histograms formed from unordered feature sets, and implicitly finds correspondences based on the finest resolution histogram cell where a pair of points first shares a bin. In practice, the induced matching approximates the optimal partial matching between sets $\mathbf{X}_i$ and $\mathbf{X}_j$

by computing the similarity:

$$\mathcal{K}(\mathbf{X}_i, \mathbf{X}_j) = \sum_{k=0}^{L} w_k \Big(\mathcal{I}_k(\mathbf{X}_i, \mathbf{X}_j) - \mathcal{I}_{k-1}(\mathbf{X}_i, \mathbf{X}_j) \Big),$$

$$\mathcal{I}_k(\mathbf{X}_i, \mathbf{X}_j) = \sum_{n=1}^{b_k} \min \Big(H_k(\mathbf{X}_i^{(n)}), H_k(\mathbf{X}_j^{(n)}) \Big), \quad (1)$$

where $H_k(\mathbf{X})$ is a histogram over the point set $\mathbf{X}$ having b_k multi-dimensional bins with sides of length 2^k, $H_k(\mathbf{X}^{(n)})$ is the count in bin n, L is the number of total resolution levels in the pyramids, and w_k reflects the similarity between points matched at level k. Setting the weights as $w_k = \frac{1}{2^k}$ corresponds to a similarity weighting inversely proportional to the bin size, i.e., the worst possible similarity of matched points. See [4] for details.

Comparing sets of image descriptors in this way provides an efficient (linear in the number of features m_i) measure of how well the two sets' features may be put into correspondence. The matching is *partial* since a subset of the sets' features may be ignored without any penalty to the matching score. This is desirable when we want to learn from unlabeled images containing multiple classes, varying backgrounds, and occlusions—cases where portions of the feature sets may be considered outliers that should not affect the matching quality.

The pairwise pyramid match affinities over feature sets serve to form an undirected, fully-connected graph over $\mathbf{U}$: nodes are images, and edges are weighted according to partial-match similarity between the images' feature sets. Within this graph, we would like to discover categories from those images with the strongest aggregate feature matchings. We seek the partitioning of the nodes that will preserve strongly connected groups while dividing nodes with minimal joint correspondences.

To this end, we employ spectral clustering and use the normalized cuts criterion developed in [11] for image segmentation. The algorithm "cuts" the nodes into disjoint sets by removing connecting edges; the optimal partitioning both minimizes the amount of dissociation between groups and maximizes the association within groups. The normalized cut dissociation measure is essentially designed to remove edges between the least similar nodes without favoring cuts that partition out small numbers of isolated points. In our case, this means enforcing that a few images that happen to have exceptional feature matchings should not be selected as categories when there exist broader range associations between feature sets.

Though minimizing the normalized cut is NP-complete, the authors of [11] provide an efficient approximate solution based on solving an eigenvalue problem,

$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{K})\mathbf{D}^{-\frac{1}{2}} x = \lambda x, \quad (2)$$

where $\mathbf{K}$ is an $N \times N$ affinity matrix over data nodes $\{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$, $\mathbf{D}$ is a diagonal matrix containing the sums of the rows of $\mathbf{K}$, and x is an indicator vector specifying the bi-partition of the graph. To form multiple partitions, recursive cuts or multiple top eigenvectors are used. Extracting the normalized cuts grouping over the pyramid match affinity matrix with entries $\mathbf{K}_{ij} = \mathcal{K}(\mathbf{X}_i, \mathbf{X}_j)$ for all images in $\mathbf{U}$ thus provides our initial set of learned categories.

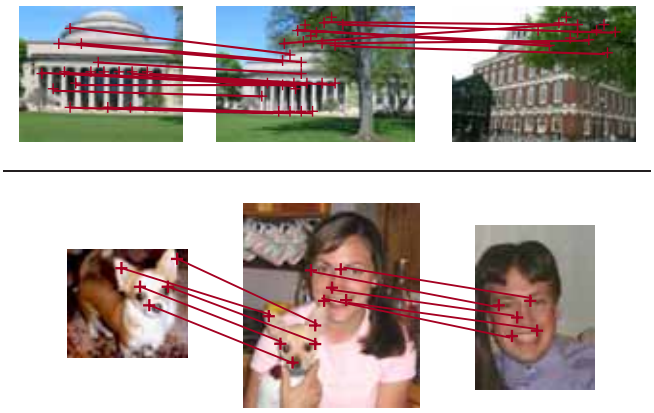

Figure 1. Graph partitioning according to partial matchings may allow problematic groups, for example when background features and foreground features find good matchings in different categories of images. In the top row, the image-to-image similarity between the right and center images may be indistinguishable from that of the center and left images, even though the right image is matching what are background features for the domed building category. In the bottom row, the presence of two categories in the center image causes it to match equally well to the images on its left and right, which contain individual instances of those categories. As a result, graph partitioning algorithms may be unable to make appropriate cuts.

This framework allows the introduction of weak semi-supervision in the form of pairwise constraints between the unlabeled images. Specifically, a user may specify "cannot-group" or "must-group" connections between any number of pairs in the data set. Following the paradigm suggested in [6], we modify the graph over $\mathbf{U}$ to incorporate this information to assist category learning: entries in the affinity matrix $\mathbf{K}$ are set to the maximal (diagonal) value for pairs that ought to be reinforced in the groupings, or set to zero for pairs that ought to be divided.

Computing affinities with the pyramid match requires time only linear in the set size, specifically $O(mL)$ for sets with $O(m)$ features and pyramids with L levels. For data sets with a large number of example sets to be clustered, we can avoid computing all $O(N^2)$ affinities and obtain a more efficient estimate of the pyramid match kernel matrix by employing the Nyström approximation technique [13].

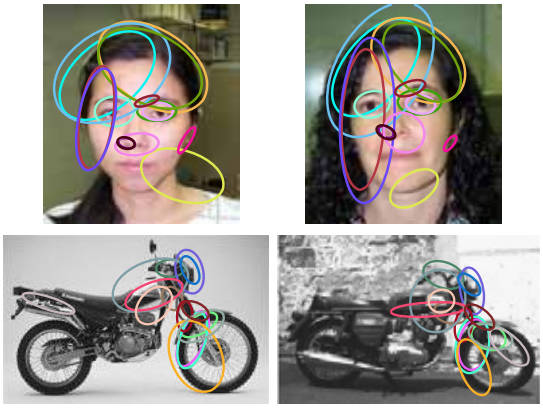

Figure 2. Examples of explicit feature correspondences extracted from a pyramid matching. Displayed here are the most confident matches found for two image pairs, as denoted by the color-coded elliptical feature regions. (This figure is best viewed in color.)

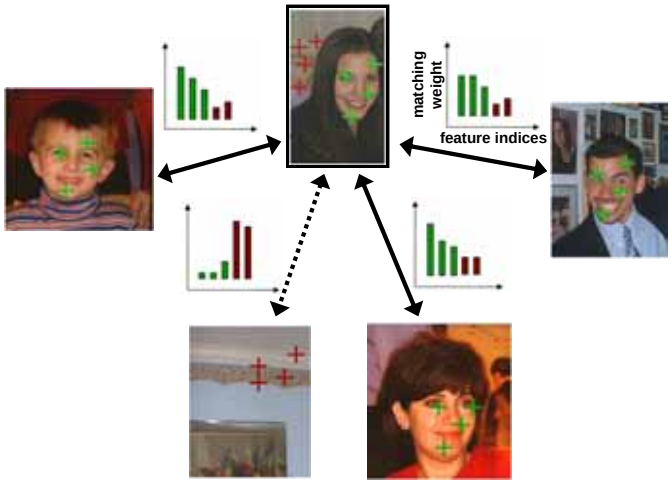

Figure 3. A schematic view of category feature mask inference. Within a single cluster, outlier images are detected by considering the typical per-image feature masks implied by which component features an image contributes to partial matchings with other members of the cluster. In this illustrative example, the similarity between the matched-feature distributions among the faces reveals the outlier non-face image, whose features happen to match the background of the top image. Shown here are the four matched-feature distributions for the top center image against the rest, with the in-mask features colored green, and non-mask features colored red. Re-weighting the correspondences according to the example's median indicator mask causes the similarity against the outlier image to be downgraded, as indicated by the dashed line. To deduce cluster outliers, feature masks are determined using all pairs in this manner. (This figure is best viewed in color.)

3.2. Inferring Category Feature Masks

Due to the nature of a partial matching, the clusters produced with normalized cuts risk containing non-homogenous members. While ignoring superfluous features without penalty to the matching similarity is desirable in the sense that it allows a degree of tolerance to clutter, outlier features, and noise, it also means that sets containing similar backgrounds may be allowed to match just as well as those containing similar objects of interest. Likewise, images containing multiple objects may find strong partial matchings with examples containing single objects from each class, thereby confounding the normalized cuts criterion in some cases (see Figure 1).

To address this, we look to the pattern of correspondences within each cluster, and leverage the information contained in the intra-cluster partial matching statistics to refine the initial grouping. The goal is to identify prototypical cluster members (or, conversely, outlier cluster members) by computing for each example the distribution of its features $\{\mathbf{f}_1, \ldots, \mathbf{f}_{m_i}\}$ that was used to form matchings with other examples within the cluster. The intuition is that we expect "inlier" images to utilize similar portions of their feature sets to form partial matches with one another, while outlier cluster members will cause most images within the cluster to contribute an inconsistent subset of features relative to their other matchings.

To apply this concept, we require the inter-feature correspondences for the pairwise partial matches within each cluster. While the method presented in [4] provides an efficient estimate of the overall matching score, it does not offer an explicit correspondence field. Here we derive a method for inducing the approximate correspondences implied by a pyramid match between two images.

The pyramid match considers feature points matched at the finest resolution pyramid level that they fall into the same bin. This means that in any bin where two point sets both contribute points, the points from the set with fewer points in that bin are certainly matched, but only some (unknown) subset of points is matched from the set having more points in that bin. If the counts are equal in a given bin, all points falling in that bin from both sets are matched to each other, in some permutation.

When computing the multi-resolution histograms for an input set of descriptors, we attach to each bin index the indices of the features that the bin spans. This allows us to trace feature matchings during the summation over histogram intersections in Eqn. 1 based on which specific points are responsible for causing an intersection at a particular level. For each input $\mathbf{X}_i$, a weighted indicator vector $\mathbf{r}_i$ of dimension m_i is maintained, i.e., $\mathbf{r}_i$ is indexed by the input's feature set. Each indicator is initialized to zero, and then at each level of the pyramid intersections it is updated to reflect the new matches formed.

The partitioning of the feature space provided by the pyramid decomposes the required matching computation into a hierarchy of smaller matchings. Upon encountering a bin with a nonzero intersection value, an explicit matching is computed between only those features from the two sets

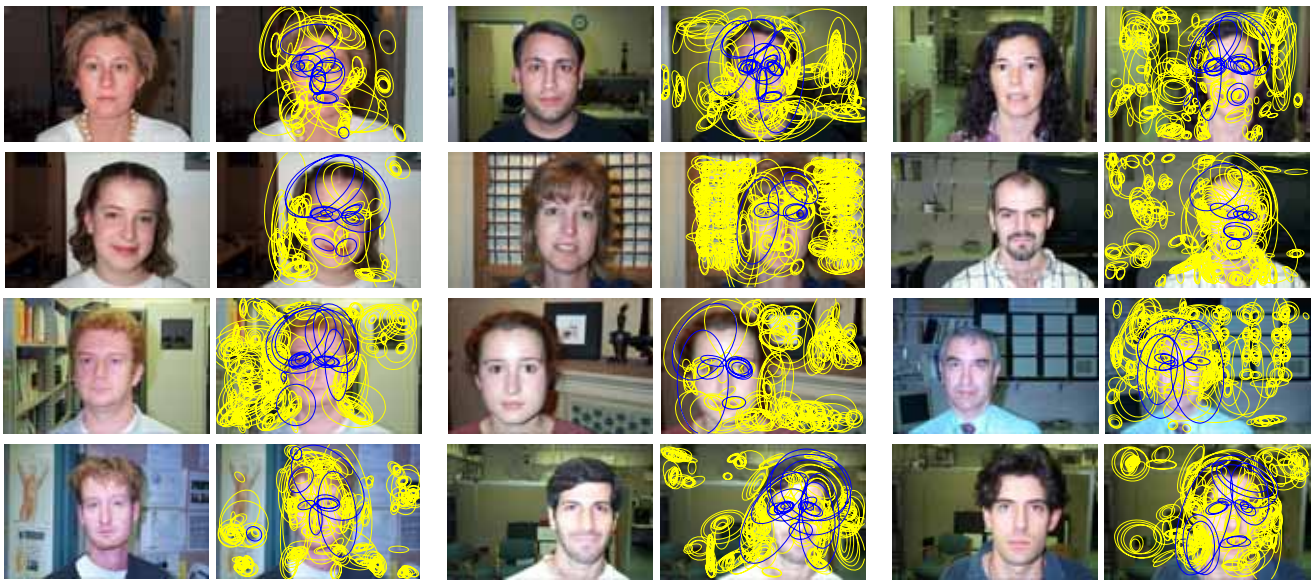

Figure 4. Inferred feature masks for a face category. The elliptical regions indicate where features were extracted from the images based on the Harris-Affine interest operator. The boundaries of these regions are color-coded in order to show which features contribute most strongly to each image's matchings against the other face images: blue (darker colored) ellipses denote the features in each image with the high weights in the mask, and yellow (light colored) ellipses denote the remaining features, which have low weights in the mask. Each entry in an inferred feature mask reflects how consistently that feature can be matched against all other images in a cluster (see Section 3.2.) These examples demonstrate how the inferred feature masks reveal which parts of the images correspond to the in-class category, and can be used to downplay the impact of background or clutter features in the matching. (This figure is best viewed in color.)

that fall into that particular bin. Given this matching, entries in the two input sets' indicator vectors corresponding to that bin's attached feature indices are recorded as the similarity between the respective matched points. All points that are used in that per-bin optimal matching are then flagged as matched and may not take part in subsequent matchings within larger bins at coarser resolutions of the pyramid.

The result is one weighted indicator vector per image, per matching comparison that reveals both which features were used in each partial match, as well as how strongly those features played a role in the total matching cost (see Figures 2 and 4). We use these indicator vectors as feature masks that designate which component features each set contributed to matchings. For each image in a cluster containing C members, a typical feature mask is computed as the median indicator vector over that image's $C-1$ within-cluster matchings.

3.3. Identifying Prototypes

To refine the groupings provided by normalized cuts clustering, the pyramid match affinities are re-computed between cluster members using the median feature masks to weight the input feature sets. That is, rather than entering unit bin counts for each feature during the pyramid computation, each feature adds a mass to the bin that is proportional to its weighting in the median feature mask for that example. Essentially this re-weights the individual feature matchings to favor those that are established between features likely to belong to the "inlier" cluster examples, and to downplay those caused by the inconsistent outlier matchings (see Figure 3). This new $C \times C$ affinity matrix is left un-normalized, since given the feature masks we no longer wish to treat small correspondence fields as being equally significant to large ones.

Having adjusted the within-cluster affinities to take the feature masks into account, we can then sort the images in each group according to how consistently they match the remainder of the group. We define the flow per example within a cluster to be the sum of its re-weighted pyramid match scores against the rest of the cluster members. Items in the cluster are then ranked according to their flow magnitudes, and examples falling within a specified top percentile of this ranking are identified as candidate prototypes. In our implementation we have evaluated the categories learned with no supervision under various settings of the percentile parameter, but we also envision allowing minimal semi-supervision at this stage, where a user could be presented with a small number of prototypes to label. Should the user disagree with the cluster labels, we could introduce link constraints into the re-weighted cluster affinity matrices here (as well as prior to performing normalized cuts) and iteratively recompute the prototypes.

The prototypes are then considered the best representatives for the particular learned category, and may be used

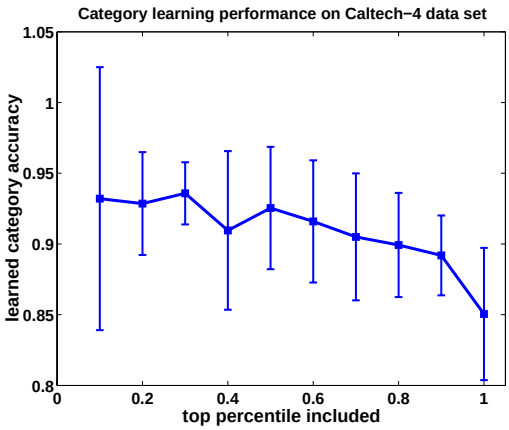

Figure 5. Accuracy of categories learned without supervision, as measured by agreement with ground truth labels. The percentiles determine the amount of prototype candidates to keep per learned class, and results shown here are averaged over 40 runs for each. The plotted points denote the mean performance for those runs and error bars denote the standard deviation. See text for details.

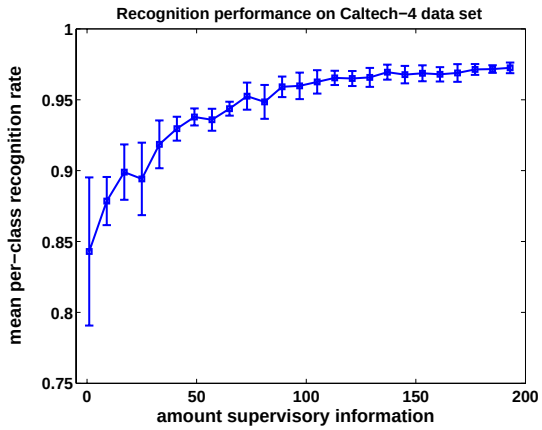

Figure 6. Recognition performance on unseen images using categories learned with varying amounts of weak semi-supervision. The horizontal axis denotes the number of (randomly chosen) "must-group" pairings provided, and the vertical axis denotes the recognition performance averaged over four classes. The plotted points are mean values and error bars are the standard deviations over 40 runs with randomly selected training/testing pools. See text for details.

to build a classifier that can predict categories for novel images. In our implementation we have chosen to use a discriminative kernel-based classifier, the support vector machine, since it can directly use the same pyramid match kernel matrix as the normalized cuts computation; however, other alternatives are equally plausible.

4. Results

In this section we present results evaluating the proposed method when applied to perform unsupervised or semi-supervised learning of object categories, and we show its ability to automatically train a classifier that can be used to predict the labels of unseen images.

We have experimented with a common benchmark data set containing four object classes, the Caltech-4 database, which is comprised of 1155 rear views of cars, 800 images of airplanes, 435 images of frontal faces, and 798 images of motorcycles. Many of the images of the airplanes and motorcycles contain white borders which we removed before any processing was done, so as to avoid inserting features that might provide either misleading or helpful cues to our algorithm. We detected salient points in the images with a Harris-Affine interest operator [9], and decomposed images into sets of SIFT features [7], scale-invariant descriptors based on histograms of oriented image gradients. More compact (10-dimensional) features were obtained from the original SIFT descriptors using PCA.

For the first experiment, we provided our method with a pool of unlabeled images containing examples from each class and requested that it learn four categories. Figure 5 summarizes the accuracy of the groupings produced as a function of the percentage of prototypes extracted from each initial normalized cuts grouping as discussed above. Accuracy is measured as the mean diagonal of a confusion matrix computed for the learned categories against the ground truth labels. For each percentile level tested, we ran the method 40 times, each time with a different random subset of 400 images from the 3188 total images, with 100 images per class. This result demonstrates the impact of the refinement mechanism to detect prototypical cluster examples based on the inferred feature masks. Figure 4 shows some examples of inferred category masks for face images with cluttered backgrounds from the Caltech data set.

We have also evaluated how the categories learned with our method will generalize to predict labels for novel images. We trained support vector machines with the pyramid match kernel using the labels produced with varying amounts of semi-supervision (see Figure 6). Recognition performance is measured as the mean diagonal of a confusion matrix computed for a total of 2788 novel test images of the four classes (ranging from about 300 to 1000 test images per class), and results are averaged over 40 runs with different randomly selected pools of 400 unlabeled "training" images. Semi-supervised constraints are of the "must-group" form between pairs of unlabeled examples, and an equal number of such constraints was randomly selected for each class from among the training pool. The results suggest that the category learning stands to gain from even rather small amounts of weak supervision.

5. Conclusions and Future Work

We have proposed a novel approach to unsupervised and semi-supervised learning of categories from inputs that are variable-sized sets of unordered features. Sets of local image features are efficiently compared in terms of partial-match correspondences between component features, forming a graph between the examples that is partitioned via spectral clustering. We have also presented modifications to an existing implicit matching kernel that allow explicit correspondence fields to be efficiently extracted, and have designed a method for inferring the typical feature mask within a learned category using these correspondences. The results indicate that reasonably accurate unsupervised recognition performance is obtainable using a very efficient method.

In our experiments, the number of groupings formed by normalized cuts was specified as the number of classes expected in the data set; however automated model selection techniques could certainly be applied (see for instance, [14]) and will be an interesting consideration for future implementations.

The idea of using within-cluster matched-feature distribution statistics to recover prototypical images is exciting to us, and appears promising. However, it does rely on the assumption that a clustering will provide an adequately strong primary category mode. It will be interesting to further explore the concept and consider ways to relax this assumption.

Another open question regarding the concepts presented is what particular types of local features and interest operators are most appropriate for discerning object categories within this framework. For instance, in our recent experiments doing supervised classification with a large set of categories, we saw that sets of features sampled densely from the images may provide more accurate categorization results [5]. Finally, although we have focused on the object recognition application here, it will be interesting to consider an image retrieval application, where the semi-supervision aspect also seems especially relevant.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24):509–522, April 2002.

[2] A. Berg, T. Berg, and J. Malik. Shape Matching and Object Recognition using Low Distortion Correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, June 2005.

[3] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Google's Image Search. In *Proceedings of the IEEE International Conference on Computer Vision*, Beijing, China, October 2005.

[4] K. Grauman and T. Darrell. The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. In *Proceedings of the IEEE International Conference on Computer Vision*, Beijing, China, October 2005.

[5] K. Grauman and T. Darrell. Pyramid Match Kernels: Discriminative Classification with Sets of Image Features. Technical Report MIT-CSAIL-TR-2006-020, MIT, March 2006.

[6] S. Kamvar, D. Klein, and C. Manning. Spectral Learning. In *Proceedings of the International Conference on Artificial Intelligence*, Acapulco, Mexico, August 2003.

[7] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, January 2004.

[8] K. Mikolajczyk and C. Schmid. Indexing Based on Scale Invariant Interest Points. In *Proceedings of the IEEE International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[9] K. Mikolajczyk and C. Schmid. Scale and Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 1(60):63–86, October 2004.

[10] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. VanGool. Modeling Scenes with Local Descriptors and Latent Aspects. In *Proceedings of the IEEE International Conference on Computer Vision*, Beijing, China, October 2005.

[11] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[12] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering Object Categories in Image Collections. In *Proceedings of the IEEE International Conference on Computer Vision*, Beijing, China, October 2005.

[13] C. Williams and M. Seeger. Using the Nystrom Method to Speed Up Kernel Machines. In *Advances in Neural Information Processing*, 2001.

[14] L. Zelnik-Manor and P. Perona. Self-Tuning Spectral Clustering. In *Advances in Neural Information Processing*, Vancouver, Canada, December 2004.