TO APPEAR AT THE 2005 IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION

Learning Hierarchical Models of Scenes, Objects, and Parts

Erik B. Sudderth, Antonio Torralba, William T. Freeman, and Alan S. Willsky
Electrical Engineering & Computer Science, Massachusetts Institute of Technology
esuddert@mit.edu, torralba@csail.mit.edu, billf@mit.edu, willsky@mit.edu

Abstract

We describe a hierarchical probabilistic model for the detection and recognition of objects in cluttered, natural scenes. The model is based on a set of parts which describe the expected appearance and position, in an object centered coordinate frame, of features detected by a low-level interest operator. Each object category then has its own distribution over these parts, which are shared between objects. We learn the parameters of this model via a Gibbs sampler which uses the graphical model's structure to analytically average over many parameters. Applied to a database of images of isolated objects, the sharing of parts among objects improves detection accuracy when few training examples are available. We also extend this hierarchical framework to scenes containing multiple objects.

1. Introduction

In this paper, we develop methods for the visual detection and recognition of object categories. We argue that multi–object recognition systems should be based on models which consider the relationships between different object categories during the training process. This approach provides several benefits. At the lowest level, significant computational savings can be achieved if different categories share a common set of features. More importantly, jointly trained recognition systems can use similarities between object categories to their advantage by learning features which lead to better generalization [4, 18]. This inter–category regularization is particularly important in the common case where few training examples are available.

In complex, natural scenes, object recognition systems can be further improved by using contextual knowledge about the objects likely to be found in a given scene, and common spatial relationships between those objects [7, 19, 20]. In this paper, we propose a hierarchical generative model for objects, the parts composing them, and the scenes surrounding them. The model, which is summarized in Figs. 1 and 5, shares information between object categories in three distinct ways. First, parts define distributions over a common low–level feature vocabularly, leading to computational savings when analyzing new images. In addition, and more unusually, objects are defined using a common set of parts. This structure leads to the discovery of parts with interesting semantic interpretations, and can improve performance when few training examples are available. Finally, object appearance information is shared between the many scenes in which that object is found.

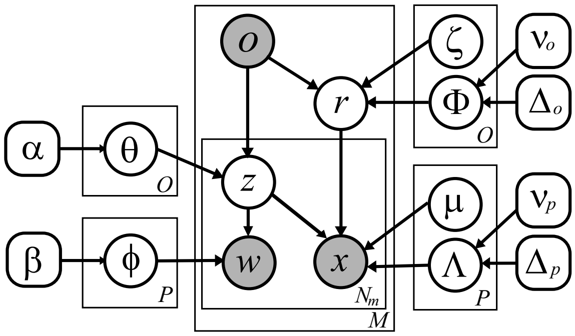

Figure 1. Graphical model describing how latent parts $\mathbf{z}$ generate the appearance $\mathbf{w}$ and position $\mathbf{x}$, relative to an image–specific reference location $\mathbf{r}$, of the features detected in an image of object $\mathbf{o}$. Boxes denote replication of the corresponding random variables: there are M images, with N_m observed features in image m.

We begin in Sec. 2 by describing our generative model for objects and parts, including a discussion of related work in the machine vision and text analysis literature. Sec. 3 then describes parameter estimation methods which combine Gibbs sampling with efficient variational approximations. In Sec. 4, we provide simulations demonstrating the potential benefits of feature sharing. We conclude in Sec. 5 with preliminary extensions of the object hierarchy to scenes containing multiple objects.

2. A Generative Model for Object Features

Our generative model for objects is summarized in the graphical model (a directed Bayesian network) of Fig. 1. The nodes of this graph represent random variables, where shaded nodes are observed during training, and rounded boxes are fixed hyperparameters. Edges encode the conditional densities underlying the generative process [12].

2.1. From Images to Features

Following [17], we represent each of our M grayscale training images by a set of SIFT descriptors [13] computed on affine covariant regions. We use K-means clustering to vector quantize these descriptors, producing a finite dictionary of F appearance patterns. This feature set provides some invariance to lighting and pose changes, and was more effective than features based on unnormalized pixel patches [21] in our experiments.

Given this feature dictionary, the i^{th} interest point in image m is described by its position x_{mi} and the best matching descriptor w_{mi}. Let $\mathbf{w}_m$ and $\mathbf{x}_m$ denote the appearance and position, respectively, of the N_m features in image m. Examples of features detected in this way are shown in Fig. 2.

2.2. Using Parts to Generate Objects

The representation of objects as a collection of spatially constrained parts has a long history in vision [8]. In the graphical model of Fig. 1, parts $\mathbf{z}$ are formalized as clusters of features that appear in similar locations, and have similar appearance. Object categories are in turn defined by a probability distribution θ specifying which parts are most likely to produce corresponding visual features.

Consider the generative process for an image of object o_m containing N_m features $(\mathbf{w}_m, \mathbf{x}_m)$. All feature positions are defined relative to an image–specific coordinate frame, or reference position, r_m. Each object category has its own Gaussian prior over reference positions:

$$p\left(r_m \mid o_m\right) = \mathcal{N}(r_m;\, \zeta_{o_m}, \Phi_{o_m}) \qquad (1)$$

To generate the i^{th} feature, we first independently sample a part z_{mi} according to an object–specific multinomial distribution θ_{o_m} over the P possible parts. Then, conditioned on the chosen part index z_{mi}, we independently sample an appearance w_{mi} and position x_{mi}:

$$p\left(w_{mi}, x_{mi} \mid z_{mi} = j, r_m\right) = \phi_j(w_{mi}) \times \mathcal{N}(x_{mi};\, r_m + \mu_j, \Lambda_j) \qquad (2)$$

Each part z is defined by a multinomial distribution ϕ_z over the F possible appearance descriptors, as well as a Gaussian distribution over feature positions. Because the mean of this Gaussian is shifted relative to r_m, we may recognize objects whose spatial translation varies from image to image.

Although we assume the collection of objects is known, the probability distributions defining this generative model must be learned from training data. The hierarchical structure allows information to be shared in two distinct ways: parts combine the same features in different spatial configurations, and objects reuse the same parts in different proportions. The learning process, as described in Sec. 3, is free to give each object category its own parts, or "borrow" parts from other objects, depending on which better explains the observed images. As we show in Sec. 4, this sharing can significantly improve detection performance.

When learning statistical models from small data sets, prior distributions play an important regularizing role [4]. To simplify the learning process, we assume that these priors have a conjugate form [9]. In particular, the multinomial distributions θ and ϕ are assigned independent, symmetric Dirichlet priors with hyperparameters α and β, respectively. The covariance matrices Λ_z of the Gaussian part position densities have inverse–Wishart priors with scale Δ_p and ν_p degrees of freedom, while the means μ_z are given noninformative priors. Similarly, the reference position's covariance prior is inverse–Wishart with hyperparameters Δ_o and ν_o.

2.3. Related Models

The graphical model of Fig. 1 was partially inspired by recently proposed models of text documents. In particular, if position variables are neglected, we recover a variant of the *author–topic model* [15], where objects correspond to authors, features to words, and parts to the latent topics underlying a given corpus. The generative aspect model, or *latent Dirichlet allocation (LDA)* [2, 10], is in turn a special case in which each document has its own topic distribution, and authors are not explicitly modeled.

LDA has been previously adapted to discover object categories from images of single objects [17], categorize natural scenes [6], and (with a slight extension) parse presegmented captioned images [1]. However, following an initial stage of low–level feature extraction [6, 17] or segmentation [1], these models ignore spatial information, treating the image as an unstructured *bag of words*. In contrast, our introduction of a reference position allows us to explicitly model the spatial locations of detected features. This extension raises additional computational issues, which we address using the EM algorithm (Sec. 3.2), and leads to improved performance in detection and recognition tasks.

When modeling a single object category, our model also shares many features with constellation models [8], particularly recent extensions which use Bayesian priors when learning from few examples [4, 5]. The principal difference is that their likelihood assumes that each part generates at most one observed feature, creating a combinatorial data association problem for which greedy approximations are needed to learn more than a few parts [11]. In contrast, our association of objects with distributions over parts leads to simple learning algorithms which scale linearly with P. In addition, by sharing parts when learning multiple object categories, we can improve generalization performance.

3. Learning Objects with Shared Parts

In this section, we derive a Gibbs sampling algorithm for learning the parameters of the hierarchical model of Fig. 1. We begin in Sec. 3.1 by assuming that all objects occur at roughly the same position in each image, so that the reference position r_m can be neglected. Many standard object recognition datasets, as well as systems which use cues such as motion to focus attention, satisfy this assumption. In Sec. 3.2, we extend the Gibbs sampler by using the EM algorithm to analytically average over the unobserved reference position. In both cases, we assume that all hyperparameters have fixed, known values, and that each training image has been labeled with the single object o_m it contains.

3.1. Monte Carlo Feature Clustering

When the reference position r_m is removed from Fig. 1, the resulting hierarchical structure is similar to the Author–Topic model, except that an additional observation (the position x_{mi}) is associated with each part indicator z_{mi}. Following [10, 15], we learn this model's parameters by Gibbs sampling the part assignments $\mathbf{z}$ using likelihoods which analytically integrate over θ, ϕ, μ, and Λ. These approximate samples from the posterior distribution $p\left(\mathbf{z} \mid \mathbf{w}, \mathbf{x}, \mathbf{o}\right)$ then provide estimates of the underlying parameters.

Let $\bar{\mathbf{z}}_{mi}$ denote the set of all part assignments excluding z_{mi}, and define $\bar{\mathbf{w}}_{mi}$ and $\bar{\mathbf{x}}_{mi}$ similarly. The Gibbs sampler iteratively fixes the part assignments $\bar{\mathbf{z}}_{mi}$ for all but one feature, and then samples a part z_{mi} for the remaining feature from the induced conditional distribution. Using the Markov properties of the graph in Fig. 1, the posterior distribution over part assignments factors as follows:

$$p\left(z_{mi} \mid \bar{\mathbf{z}}_{mi}, \mathbf{w}, \mathbf{x}, \mathbf{o}\right) \;\propto\; p\left(z_{mi} \mid \bar{\mathbf{z}}_{mi}, o_m\right) \times \\ p\left(w_{mi} \mid \mathbf{z}, \bar{\mathbf{w}}_{mi}\right) p\left(x_{mi} \mid \mathbf{z}, \bar{\mathbf{x}}_{mi}\right) \quad (3)$$

Let n_{kj}^{FP} denote the number of times feature k is assigned to part j by $\bar{\mathbf{z}}_{mi}$, and $n_{\ell j}^{OP}$ the number of assignments of part j to object ℓ. Using standard Dirichlet integral formulas [9], the first two terms of eq. (3) can be written as

$$p\left(z_{mi} = j \mid \bar{\mathbf{z}}_{mi}, o_m = \ell\right) = \frac{n_{\ell j}^{OP} + \alpha}{\sum_{j'} n_{\ell j'}^{OP} + P\alpha} \quad (4)$$

$$p\left(w_{mi} = k \mid z_{mi} = j, \bar{\mathbf{z}}_{mi}, \bar{\mathbf{w}}_{mi}\right) = \frac{n_{kj}^{FP} + \beta}{\sum_{k'} n_{k'j}^{FP} + F\beta} \quad (5)$$

Note that these probabilities are simply the raw proportions defined by the part assignments $\bar{\mathbf{z}}_{mi}$, regularized by the "pseudocounts" contributed by the Dirichlet priors.

Given the current part assignments and the inverse–Wishart prior, the posterior distribution over x_{mi} is multivariate Student-t with $(n_j^P + \nu_p - 1)$ degrees of freedom [9], where n_j^P is the total number of features of any appearance assigned to part j. For the parameters used in our experiments, this likelihood is very closely approximated by a moment–matched Gaussian:

$$p\left(x_{mi} \mid z_{mi} = j, \bar{\mathbf{z}}_{mi}, \bar{\mathbf{x}}_{mi}\right) \approx \mathcal{N}\left(x_{mi};\, \hat{\mu}_j, \hat{\Lambda}_j\right) \quad (6)$$

$$\hat{\mu}_j = \frac{1}{n_j^P} \sum_{m=1}^{M} \sum_{k \mid z_{mk}=j} x_{mk} \qquad \delta_j = \frac{n_j^P + 1}{n_j^P (n_j^P + \nu_p - 3)}$$

$$\hat{\Lambda}_j = \delta_j \left(\Delta_p + \sum_{m=1}^{M} \sum_{k \mid z_{mk}=j} (x_{mk} - \hat{\mu}_j)(x_{mk} - \hat{\mu}_j)^T \right)$$

All sums in eq. (6) exclude the feature x_{mi} whose assignment z_{mi} is being resampled. Intuitively, this Gaussian exactly matches the sample mean of the features associated with that part, while regularizing the sample covariance by the inverse–Wishart prior.

By combining eqs. (4, 5, 6), we may evaluate eq. (3) for each of the P candidate assignments of z_{mi}, and perform the Gibbs sampling iterations. By caching and iteratively updating the counts used in eqs. (4, 5), as well as sums and outer products of the positions of features assigned to each part, this sampling update can be performed in $\mathcal{O}(P)$ operations. Given a training set of M images, each containing N features, the total cost of a Gibbs sampling update of every feature assignment is $\mathcal{O}(NMP)$.

3.2. Inferring Reference Positions

We now consider the full graphical model of Fig. 1, including the reference position r_m. In this case, the conditional distribution needed by the Gibbs sampler is given by

$$p\left(z_{mi} \mid \bar{\mathbf{z}}_{mi}, \mathbf{w}, \mathbf{x}, \mathbf{o}\right) \;\propto\; p\left(z_{mi} \mid \bar{\mathbf{z}}_{mi}, o_m\right) \times \\ p\left(w_{mi} \mid \mathbf{z}, \bar{\mathbf{w}}_{mi}\right) p\left(x_{mi} \mid \mathbf{z}, \bar{\mathbf{x}}_{mi}, \mathbf{o}\right) \quad (7)$$

While the first two terms are unchanged from eqs. (4, 5), the uncertainty in the Gaussian parameters (ζ, Φ) causes the position likelihood to depend on the reference position estimates, and hence the object labels, of all training images. In addition, because of coupling between the parameters of the part and reference position Gaussians, this likelihood can no longer be expressed in the closed form of eq. (6).

While computing the position likelihood of eq. (7) is intractable, it would be simple if the Gaussian prior parameters were known. We therefore propose to approximate this likelihood by finding the mode of the conditional distribution of these parameters:

$$\left(\hat{\mu}, \hat{\Lambda}, \hat{\zeta}, \hat{\Phi}\right) = \arg\max\; p\left(\mu, \Lambda, \zeta, \Phi \mid \bar{\mathbf{z}}_{mi}, \bar{\mathbf{x}}_{mi}, \mathbf{o}\right) \quad (8)$$

Because the reference positions are unobserved, we use the EM algorithm [9, 14] to iteratively compute this mode.

In the E-step, we fix the current values for the mean and covariance parameters of eq. (8), and average over the unobserved reference position r_m in each training image m. Given these parameters, the reference positions have Gaussian posterior distributions, with covariance and mean

$$R_m = \left(\hat{\Phi}_{o_m}^{-1} + \sum_{j=1}^{P} \sum_{k \mid z_{mk}=j} \hat{\Lambda}_j^{-1} \right)^{-1} \quad (9)$$

$$\hat{r}_m = \hat{\zeta}_{o_m} + R_m \left(\sum_{j=1}^{P} \sum_{k \mid z_{mk}=j} \hat{\Lambda}_j^{-1} \left(x_{mk} - \hat{\mu}_j - \hat{\zeta}_{o_m} \right) \right)$$

The sums in eq. (9) only include the feature positions from the corresponding image m. These expected reference positions define a lower bound on the likelihood, which is maximized by the M-step. Given M_ℓ images of object ℓ, the maximizing reference position parameters equal

$$\hat{\zeta}_\ell = \frac{1}{M_\ell} \sum_{m|o_m=\ell} \hat{r}_m \qquad\qquad \delta_\ell = \frac{1}{M_\ell + \nu_o + 3} \tag{10}$$

$$\hat{\Phi}_\ell = \delta_\ell \left(\Delta_o + \sum_{m|o_m=\ell} R_m + (\hat{r}_m - \hat{\zeta}_\ell)(\hat{r}_m - \hat{\zeta}_\ell)^T \right)$$

The part position parameters are similarly updated as

$$\begin{aligned} \hat{\mu}_j &= \frac{1}{n_j^P} \sum_{m=1}^{M} \sum_{k|z_{mk}=j} (x_{mk} - \hat{r}_m) \\ \hat{\Lambda}_j &= \frac{1}{n_j^P + \nu_p + 3} \left(\Delta_p + \sum_{m=1}^{M} \hat{\Lambda}_{jm} \right) \qquad (11) \\ \hat{\Lambda}_{jm} &= \sum_{k|z_{mk}=j} R_m + (x_{mk} - \hat{\mu}_j - \hat{r}_m)(x_{mk} - \hat{\mu}_j - \hat{r}_m)^T \end{aligned}$$

Note that the updates of eq. (11) are similar to the moment matching of eq. (6), except that parts are translated by the current expected reference position in each image.

We apply these EM updates between every Gibbs sampling operation. Because the posterior mode is not dramatically changed by the reassignment of a single feature, only a single EM iteration per sample is needed for accurate mode tracking. Conditioned on the parameter estimates produced by the M-step, the reference position r_m has a Gaussian distribution with mean and covariance as in eq. (9). The feature position likelihood then has the following closed form:

$$\begin{aligned} p\left(x_{mi} \mid z_{mi} = j, \bar{\mathbf{z}}_{mi}, \bar{\mathbf{x}}_{mi}, \mathbf{o}\right) &= \\ \mathcal{N}(x_{mi};\, \hat{r}_m + \mu_j, R_m + \Lambda_j)& \qquad (12) \end{aligned}$$

This expression is used in eq. (7) to evaluate the probabilities for each Gibbs sampling operation.

Direct implementation of these EM updates requires $\mathcal{O}(MP)$ operations per iteration due to the coupling between the reference positions and parts. However, we may reduce the cost of each iteration to $\mathcal{O}(P)$ using incremental EM updates [14]. In particular, when sampling a part assignment for image m, we fix the expectations of eq. (9) for all reference positions except r_m. By caching statistics of the other reference position estimates, the M–step (eqs. (10, 11)) may also be performed efficiently. Although we no longer find the exact posterior mode, the dependencies of the reference positions in other images on $\mathbf{z}_m$ are very weak, so this approximation is extremely accurate. Empirically, incremental updates produce dramatic computational gains with negligible loss of sampling accuracy.

3.3. Likelihoods for Object Detection

To use the hierarchical model for detection or recognition, we must compute the likelihood that a test image t, with features $(\mathbf{w}_t, \mathbf{x}_t)$, is generated by each candidate object category o. Because each image's features are independently sampled from a common parameter set, we have

$$p\left(\mathbf{w}_t, \mathbf{x}_t \mid o, \mathcal{M}\right) = \int p\left(\mathbf{w}_t, \mathbf{x}_t \mid o, \Theta\right) p\left(\Theta \mid \mathcal{M}\right) d\Theta$$

In this expression, $\mathcal{M}$ denotes the set of training images, and $\Theta = (\theta, \phi, \mu, \Lambda, \zeta, \Phi)$ the model parameters. The sequence of part assignments produced by the Gibbs sampler provides samples $\mathbf{z}^{(s)}$ approximately distributed according to $p\left(\mathbf{z} \mid \mathcal{M}\right)$. Given a set of S samples, we approximate the test image likelihood as

$$p\left(\mathbf{w}_t, \mathbf{x}_t \mid o, \mathcal{M}\right) \approx \frac{1}{S} \sum_{s=1}^{S} p\left(\mathbf{w}_t, \mathbf{x}_t \mid o, \hat{\Theta}^{(s)}\right) \tag{13}$$

where $\hat{\Theta}^{(s)}$ denotes the approximate modes of the posterior distribution over parameters computed using $\mathbf{z}^{(s)}$ in eqs. (4, 5, 6, 10, 11).

When the reference position is neglected, the image features are independent conditioned on the model parameters:

$$\begin{aligned} p\left(\mathbf{w}_t, \mathbf{x}_t \mid o, \hat{\Theta}^{(s)}\right) &= \\ \prod_{i=1}^{N_t} \sum_{j=1}^{P} \hat{\theta}_o(j)\, \hat{\phi}_j(w_{ti})\, \mathcal{N}\left(x_{ti};\, \hat{\mu}_j, \hat{\Lambda}_j\right)& \qquad (14) \end{aligned}$$

This expression calculates the likelihood of N_t features in $\mathcal{O}(N_t P)$ operations. To account for the reference position, we first run the Gibbs sampling updates on the test image features. The EM estimates of Sec. 3.2 then provide a reference position estimate which can be combined with the likelihood of eq. (12) to evaluate eq. (14).

4. Object Categorization Experiments

To explore the advantages of sharing parts among objects, we consider a collection of 16 categories with noticeable visual similarities. Fig. 2 shows images from each category, which can be divided into three groups: seven animal faces, five animal profiles, and four wheeled vehicles. As object recognition systems scale to hundreds or thousands of categories, the inter–category similarities exhibited by this dataset will become increasingly common.

4.1. Learning Shared Parts

Given 30 training examples from each of the 16 categories, we constructed a feature appearance dictionary with $F = 600$ words, and used Gibbs sampling (Sec. 3.1) to fit a model with 32 shared parts. Because the database images had been manually aligned, the EM likelihood updates

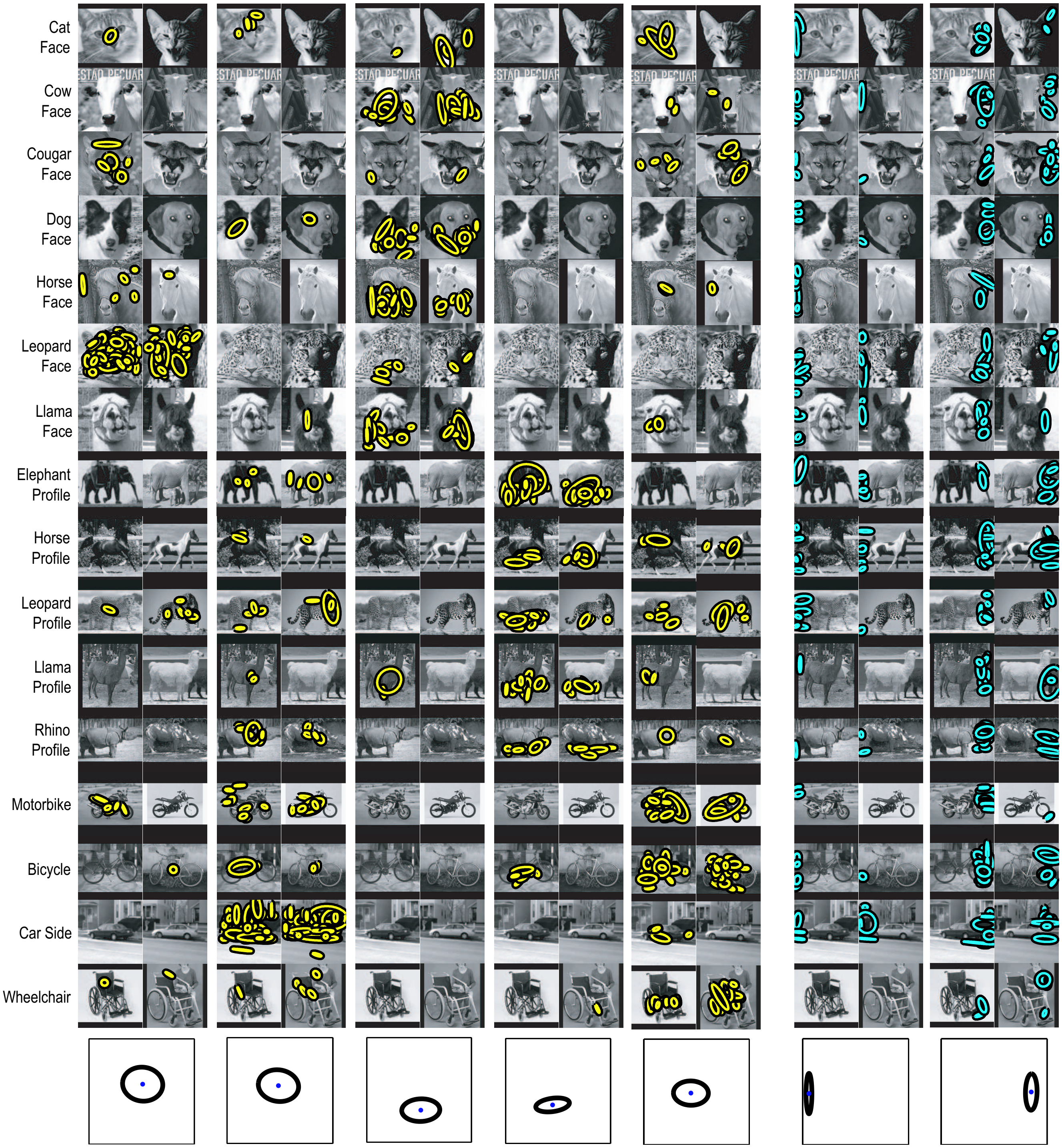

Figure 2. Seven of the 32 shared parts (columns) learned by a hierarchical model of 16 object categories (rows). Using two images from each category, we show the affinely adapted features with the highest posterior probability of being generated by each part. For comparison, we show the five parts which are specialized to the fewest object categories (left, yellow), as well as two of the most widely shared parts (right, cyan), which seem to model background clutter. The bottom row plots the Gaussian position densities corresponding to each part. These object categories combine the results of web searches with images from the CalTech 101 [5], Weizmann Institute [21], and MIT-CSAIL [18, 19] databases. Including a complementary background category, there are a total of 1,879 images used for training and testing, with at least 40 images for each category.

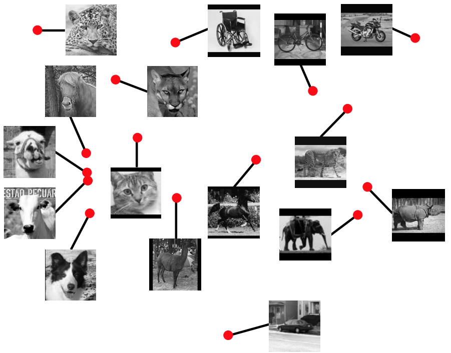

Figure 3. MDS embedding of pairwise distances between the learned part distributions for 16 object categories.

of Sec. 3.2 were not needed. For our Matlab implementation, each sampling iteration requires roughly 0.1 seconds per training image. The learning procedure showed little sensitivity to the part distribution hyperparameters, which were set to provide a weak ($\nu_p = 6$) bias towards moderate covariances and sparse ($\beta = 0.1$) appearance densities. The object–specific part distribution hyperparameter, α, was set via cross–validation as discussed below.

Following 200 iterations of the Gibbs sampler, we used the final assignments $\mathbf{z}$ to estimate each part's posterior distribution over feature appearance and position (Sec. 3.3). In Fig. 2, we visualize these distributions for seven parts. Only two parts seem specialized to a single category: a spotted texture part used by the "leopard face" category, and another part devoted to the extremely well aligned "side car" category. The next three parts model features from animal mouths, animal legs, and vehicles, respectively. We also show two of several parts which seem to model background clutter around image boundaries, and are widely shared between categories.

To further investigate these shared parts, we used the symmetrized KL divergence [15] to compute a distance between all pairs of object–specific part distributions:

$$D\left(\theta_k, \theta_\ell\right) = \sum_{j=1}^{P} \theta_k(j) \log \frac{\theta_k(j)}{\theta_\ell(j)} + \theta_\ell(j) \log \frac{\theta_\ell(j)}{\theta_k(j)} \quad (15)$$

Fig. 3 shows the two–dimensional embedding of these distances produced by multidimensional scaling (MDS) [16]. Except for cars, these part distances seem to closely match our own intuitive notions of category similarity.

4.2. Detection and Recognition

To evaluate our model, we consider two sets of experiments. In the detection task, we use 100 training images to learn an 8-part background appearance model, and then use probabilities computed as in Sec. 3.3 to classify test images as object or background. To facilitate comparisons, we also consider a recognition task in which test images are classified as either their true category, or one of the 15 other categories. For both tasks, we compare our *shared* model of all object categories to a set of 16 *unshared* models trained on individual categories. We also consider versions of both models which neglect the spatial location of features, as in recent "bag of keypoints" approaches [3, 17]. Performance curves average over three randomly chosen training sets of the given size, and use all other images for testing.

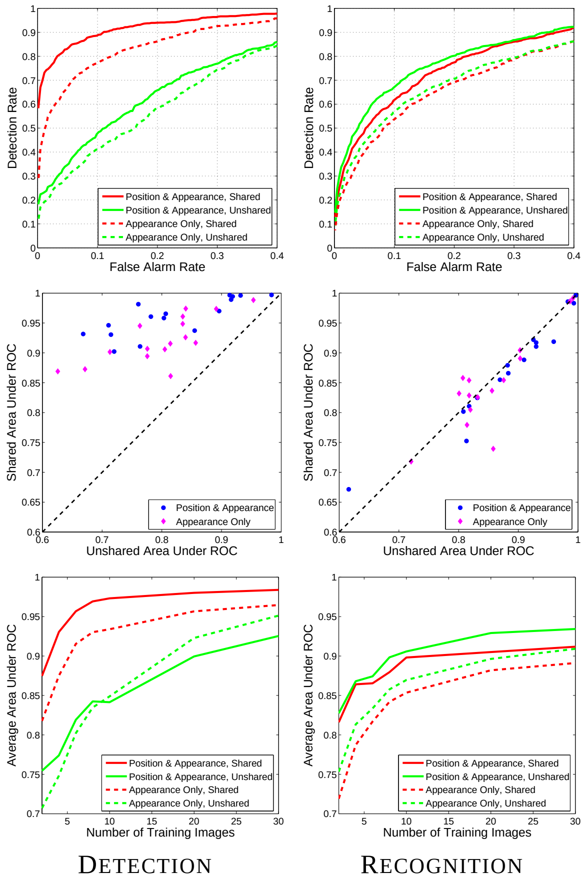

Figure 4. Performance for the tasks of detection (left) and recognition (right) of 16 object categories. TOP: **Average of ROC curves across all categories (6 training images).** MIDDLE: **Scatter plot of areas under ROC curves for the shared and unshared models of each category (6 training images).** BOTTOM: **Area under average ROC curves for different numbers of training images per category.**

As shown in Fig. 4, we find that shared parts lead to consistent, significant improvements in detection performance. These improvements are greatest when fewer than 10 training examples per category are available. For the recognition task, the shared and unshared models perform similarly, with the shared model becoming slightly less effective when many training examples are available. Confusion matrices (not shown) confirm that this slight performance degradation is produced by pairs of categories with very similar part distributions (see Fig. 3). For both tasks, feature po-

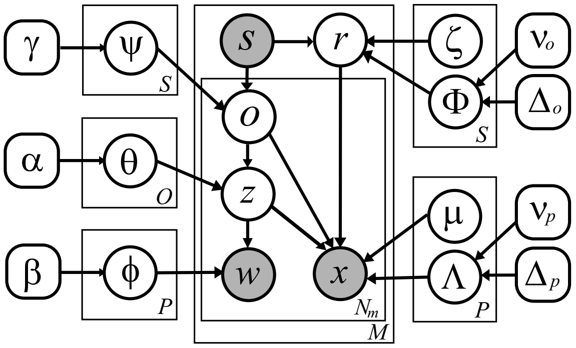

Figure 5. Extension of the graphical model of Fig. 1 to describe scenes. Each image is now associated with a scene s containing multiple objects, and r is a vector defining a reference position for each object. Each observed feature is produced by sampling an object o, a part z, and then an appearance w and position x.

sitions contain important information, and neglecting them reduces performance. Note, however, that for detection, a shared model based solely on appearance outperforms an unshared model of position and appearance.

The performance of the hierarchical model is somewhat affected by the chosen number of parts. In Fig. 4, we consider models with 2 parts per category (32 shared parts versus 16 unshared 2-part models). Using more parts slightly increases unshared detection performance (at greater computational cost), but the results remain qualitatively similar. The Dirichlet part distribution prior also has an interesting effect on shared performance with few training images. Small α values reduce sharing and slightly increase recognition performance, while large α values increase sharing, leading to improved detection. Cross–validation experiments indicated that setting $\alpha = 10/P$ provided a good compromise.

5. Hierarchical Models for Scenes

In Fig. 5, we show how the generative model of Fig. 1 may be extended to describe scenes s containing multiple objects. We specify the scene s_m associated with image m as a collection of a fixed set of O_m objects (for example, one car and two pedestrians). Conditioned on s_m, one of S possible scenes, the $2O_m$–dimensional reference vector r_m provides a reference position for each object. By learning a scene–specific prior distribution $\mathcal{N}(r_m;\, \zeta_s, \Phi_s)$, we model correlations between the locations of observed objects.

Each scene s is also associated with a discrete distribution ψ_s, which specifies the proportion of observed features generated by each object in the scene. Given s_m, for each observed feature we first sample an object indicator $o_{mi} \sim \psi_{s_m}$. Then, conditioned on this object, a part z_{mi} and feature appearance w_{mi} are sampled exactly as in Sec. 2.2. The feature position x_{mi} is then sampled from the chosen part, relative to the reference position for object o_{mi}:

$$p\,(x_{mi} \mid z_{mi} = j, o_{mi} = \ell, r_m) = \mathcal{N}(x_{mi};\, r_{m\ell} + \mu_j, \Lambda_j)$$

Here, $r_{m\ell}$ is the subvector of r_m corresponding to object ℓ.

5.1. Learning Object Relationships

Learning and inference in the scene–object–part hierarchy are natural extensions of the algorithms in Sec. 3. If the reference position is neglected, we may directly adapt the Gibbs sampler of [15], considering the combination of all pairs of values of (o_{mi}, z_{mi}) and resampling them as a block given $(\bar{\mathbf{o}}_{mi}, \bar{\mathbf{z}}_{mi})$. The reference position for scenes may also be handled with a straightforward extension of the EM updates of Sec. 3.2. The E-step now computes a $2O_m$–dimensional Gaussian distribution, in which the current assignments of features to objects act as observations of subvectors of r_m. In the M-step, the reference parameter updates are essentially identical to eq. (10), while the part parameter updates modify eq. (11) to estimate the expected deviation of each feature x_{mi} from the reference position of the currently associated object o_{mi}.

5.2. Street Scenes

To demonstrate the potential of our hierarchical scene model, we consider a simple street scene containing three "objects": buildings, cars, and roads. We use 72 images from the MIT–CSAIL database [18, 19], normalized so that cars are at comparable scales. Of these images, 26 had labels for all three categories, while the remainder only had car labels (see Fig. 6 for examples). Note that it is straightforward to incorporate such semi–supervised data into the Gibbs sampler by fixing the object labels for segmented features, and sampling the remaining labels.

We used 40 training images, and 100 Gibbs sampling iterations with reference position EM updates, to learn the parameters of a scene model with 6 shared parts (see Sec. 5.1). We then use this model to estimate the object category most likely to have generated each test image feature, producing the feature segmentations shown in Fig. 6. The model typically works well with images containing a single car (top example), except when lighting affects the low–level feature extraction (bottom example). The two middle examples have multiple cars, and thus violate our model's assumption that a single reference position explains all observed car features. In such cases, the posterior distribution over the car's reference position is spread between the vehicles, and nearby background clutter is mislabeled. Despite this issue, we find that our model of spatial scene structure outperforms a corresponding author–topic model neglecting feature positions (see the ROC curves of Fig. 6).

6. Discussion

We have described a hierarchical model for scenes, objects, and parts which shares information at several levels

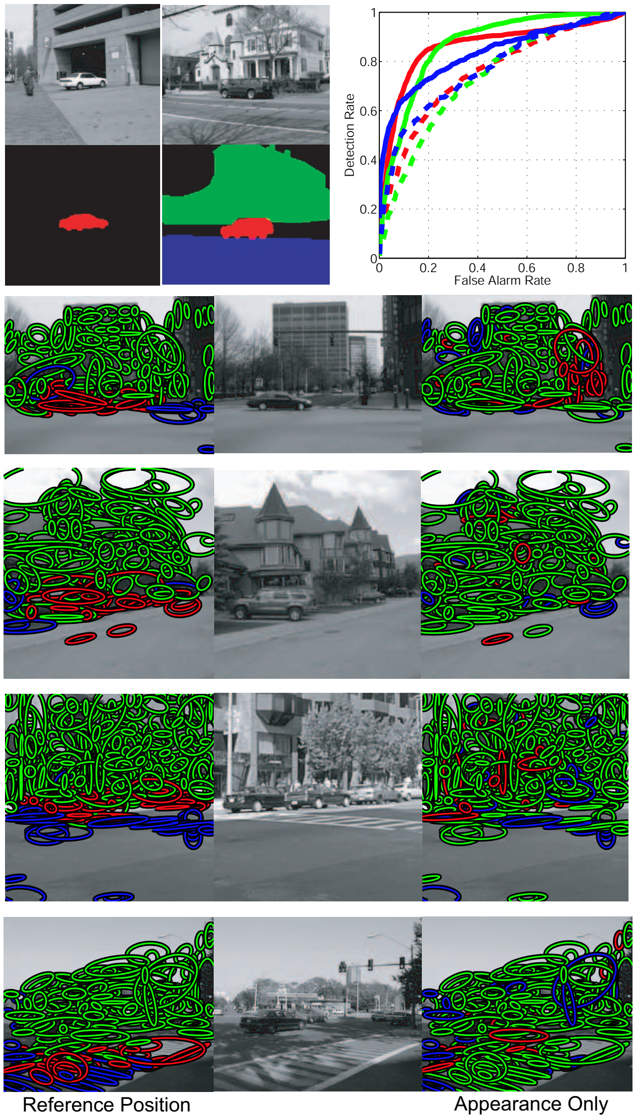

Figure 6. Feature segmentation results for a model of street scenes containing buildings (green), cars (red), and road (blue). TOP LEFT: **Two sample training images with corresponding labelings.** TOP RIGHT: **ROC curves for the detection of three color-coded object categories, using the full model with reference positions (solid), and a “bag of words” model based solely on feature appearance (dashed).** BOTTOM: **Four test images with features colored accorded to the most likely category for the reference position (left) and appearance (right) models.**

when training multiple object classifiers. Our experimental results confirm the importance of spatial structure to visual recognition problems, and show that sharing can provide significant benefits when learning from few examples. We are currently exploring richer families of scene models which allow the number of objects in each image, and parts composing each object, to be automatically inferred.

Acknowledgments

Funding provided by the National Geospatial-Intelligence Agency NEGI-1582-04-0004, the National Science Foundation NSF-IIS-0413232, the ARDA VACE program, and a BAE Systems grant.

References

[1] K. Barnard et al. Matching words and pictures. *JMLR*, 3:1107–1135, 2003.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[3] G. Csurka et al. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

[4] L. Fei-Fei, R. Fergus, and P. Perona. A Bayesian approach to unsupervised one-shot learning of object categories. In *ICCV*, volume 2, pages 1134–1141, 2003.

[5] 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 *CVPR Workshop on Generative Model Based Vision*, 2004.

[6] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *CVPR*, volume 2, pages 524–531, 2005.

[7] M. Fink and P. Perona. Mutual boosting for contextual inference. In *NIPS 16*. MIT Press, 2004.

[8] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Comp.*, 22(1):67–92, Jan. 1973.

[9] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 2004.

[10] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proc. Nat. Acad. Sci.*, 101:5228–5235, 2004.

[11] S. Helmer and D. G. Lowe. Object class recognition with many local features. In *CVPR Workshop on Generative Model Based Vision*, 2004.

[12] M. I. Jordan. Graphical models. *Statistical Science*, 19:140–155, 2004.

[13] D. G. Lowe. Distinctive image features from scale–invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[14] R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 355–368. MIT Press, 1999.

[15] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *UAI 20*, pages 487–494, 2004.

[16] R. N. Shepard. Multidimensional scaling, tree-fitting, and clustering. *Science*, 210:390–398, Oct. 1980.

[17] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. AI Memo AIM-2005-05, MIT, Feb. 2005.

[18] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: Efficient boosting procedures for multiclass object detection. In *CVPR*, pages 762–769, 2004.

[19] A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *NIPS 17*. MIT Press, 2005.

[20] Z. Tu, X. Chen, A. L. Yuille, and S. C. Zhu. Image parsing: Unifying segmentation, detection, and recognition. In *ICCV*, volume 1, pages 18–25, 2003.

[21] S. Ullman, M. Vidal-Naquet, and E. Sali. Visual features of intermediate complexity and their use in classification. *Nature Neur.*, 5(7):682–687, July 2002.