Learning Image Attributes using the Indian Buffet Process

Soravit Changpinyo
Department of Computer Science
Brown University
Providence, RI 02912
schangpi@cs.brown.edu

Erik B. Sudderth
Department of Computer Science
Brown University
Providence, RI 02912
sudderth@cs.brown.edu

Abstract

In the domain of object recognition and image classification, a recent trend is to use image properties or attributes to represent the images. Most of the proposed models in the past require that the number of attributes and attribute semantics be specified in advance. In this paper, we propose a generative model for image attributes that combine attribute-based vision models and feature-based nonparamatric models. We learn the model using Gibbs sampling. Qualitatively, we demonstrate the learned attributes of images in three categories. Quantitatively, we show that our model outperforms simple baseline methods in image retrieval and transfer learning tasks.

1 Introduction

Object recognition is an important area in computer vision. Most work in object recognition traditionally focuses on finding object categories. However, simply naming a specific object does not generalize well across categories. Some recent work [6, 26] proposes transitioning the goal of object recognition from *naming to describing*. In particular, objects can also be described in terms of certain properties or attributes. For example, we can describe a dog with attributes "brown," "hairy" and "has legs," etc. This attribute-centric approach attracts researchers for several reasons. First, it improves the degree to which machines perceive visual objects. The ability of machines to *describe* objects provides useful applications such as image search engines that perform well on specific queries. Additionally, attributes are often shared by different objects. Therefore, those attributes provide useful information for organizing collections of images. Moreover, in recognition tasks, the knowledge about attributes allows part of the learning task to be shared amoung categories. This cross-category generalization has a computational advantage. For example, Torralba et al. [24] employ a joint boosting procedure by finding common features that can be shared across object classes. By training classifiers jointly rather than independently, they discover that the required number of features grows sublinearly with the number of classes. Another advantage of having generic features is that those features are necessary when few or no training examples are available. Some recent work on attribute-based transfer learning [17] shows that attibutes can be used to build a learning object detection system that requires one or no training images of the target classes.

In almost all proposed attributed learning algorithms and attributed-based object detection frameworks, attributes must be defined a priori. Some other work [2, 27] takes advantage of natural language descriptions associated with images as a means to define their attributes. In this paper, we attempt to learn attributes in a completely unsupervised way from any given set of images. We borrow a data-driven, nonparametric Bayesian statistical method called the infinite sparse factor analysis, which is a linear transformation method in which the desired representation of multivariate data is the one that minimizes the statistical dependence of the components of the representation. In a sparse implementation, we allow the choice of whether a component is active for a data point. In addition, to allow an infinite number of components, we place a prior on a binary matrix of latent attributes using a nonparametric prior called the Indian buffet process (IBP).

The rest of the paper is organized as follows. Section 2 reviews attribute-based vision models. Section 3 reviews feature-based Bayasian nonparametric models. Section $4-5$ presents our approach to learning image attributes according to feature-based nonparametric models, as well as the learned attributes of the images from various categories. In Section 6, we evaluate our models on image retrieval and transfer learning tasks. We conclude and propose some future work in Section 7.

2 Attribute-based Vision Models

Attributed-based representaions of objects have been a topic of interest for the past several years. Researchers have proposed several attributed-based vision models to solve various tasks, including object detection/recognition [24, 27, 5], face verification [16], action recognition [18], transfer learning [17, 28, 21, 20], and image retrieval [15, 25, 9]. In this section, we briefly review related work on attribute-based vision models and their applications.

A majority of work on attribute learning collects image examplars containing or lacking a specific attribute and trains classifiers on those images to predict if new images contain such attribute. For instance, Kumar et al. [16] train classifiers for two types of attributes: one corresponding to visual appearance such as "white" or "chubby" and the other corresponding to the similarity of faces, or region of faces, to specific reference people. These attributes are then used to perform face verification. Farhardi et al. [5] train detectors for parts and superordinate categories and use the output of the classifers to vote for an object. For example, "leg," "head" and "dog" detectors would vote for the object "four-legged animal." In their framework, attributes used for voting must be semantic and must generalize well to other images of similar categories.

Another line of work involves probabilistic generative models designed for specific types of attributes. Ferrari and Zisserman [10] see attributes as patterns of segments. They propose two probabilistic generative models, one for simple attributes such as color and the other for complex attributes whose basic element consists of two segments. The model is then learned in a discriminative fashion. This approach is more specialized than our approach, as it is designed to detect specific attribute types, including "dot," "stripe," and "checkerboard."

In contrast to our attribute learning approach, all such methods above manually define attributes a priori. Our approach does not aim to specially capture any particular kinds of attributes. Rather, we intend to find, with minimum supervision, a set of attributes that are useful in some sense.

As for the unsupervised related work, the work of Sudderth et al. [22] is along the same line as our work. In their work, they propose probabilistic models for objects and the parts composing them based on another nonparametric model called heirarchical Dirichlet processes. They also employ transformed Dirichlet processes to model uncertainty in the number of object categories and object instances.

Another attempt to learn image attributes without labeled training data comes from the work of Berg et al. [2]. In their framework, text associated with each image is needed to group images together. Then, within each group, attribute candidates are picked based on visualness and characterized based on localizability and type. Our approach differs from this approach because we only consider images without text descriptions.

Recently, Liu et al. [18] propose a way to automatically learn *data-driven* attributes in addition to the ones defined a priori. Their approach clusters low-level features while maximizing the system information gain. Specifically, they merge two clusters that have a high degree of mutual information whenever they can. This method, in a sense, is similar to our approach, but it is done in a discriminative fashion and the number of clusters needs to be specified in advance.

3 Feature-based Bayesian Nonparametric Models

3.1 The Indian Buffet Process

The Indian buffet process (IBP) [11] is a distribution over equivalence classes of binary matrices with a finite number of rows and an unbounded number of columns. It can be used for nonparametric latent feature modeling in which rows correspond to data points and columns correspond to latent features. The probability of any particular matrix is given by

$$P([Z]) = \frac{\alpha^K}{\Pi_{h\in\{0,1\}^N\backslash\mathbf{0}} K_h!} \exp\{-\alpha H_N\} \prod_{k=1}^{K} \frac{(N-m_k)!(m_k-1)!}{N!}, \tag{1}$$

where K is the number of nonzero columns in Z, m_k is the number of ones in column k of Z, H_N is the N^{th} harmonic number, and K_h is the number of occurences of the non-zero binary vector of h among columns in Z.

A binary matrix can be sampled by the following metaphor: the rows represent customers and the columns represent dishes in an infinitely long buffet. The first customer takes the first Poisson(α) dishes. For the i^{th} customer, he takes each previously sampled dish k with probability $\frac{m_k}{i}$, as well as Poisson($\frac{\alpha}{i}$) new dishes. Figure 1 shows a binary matrix generated by the IBP [11].

One problem with this IBP model is that the parameter α controls both the number of features per object and the total number of features. To address this problem, Ghahramani et al. [11] propose a two-parameter generalization of the model that has an additional parameter β. The stochastic process described above can be modified by having instead the i^{th} customer take each previously sampled dish k with probability $\frac{m_k}{\beta+i-1}$ and try Poisson($\frac{\alpha\beta}{\beta+i-1}$) new dishes. The distribution of $[Z]$ then becomes

$$P([Z]) = \frac{(\alpha\beta)^K}{\Pi_{h\in\{0,1\}^N\backslash\mathbf{0}} K_h!} \exp\{-\alpha H_N(\beta)\} \prod_{k=1}^{K} B(m_k, N-m_k+\beta), \tag{2}$$

where $H_N(\beta) = \sum_{i=1}^{N} \frac{\beta}{\beta+i-1}$ and B is the beta function $B(r,s) = \frac{\Gamma(s)\Gamma(s)}{\Gamma(r+s)}$

Note that when $\beta = 1$, we recover the one-parameter IBP.

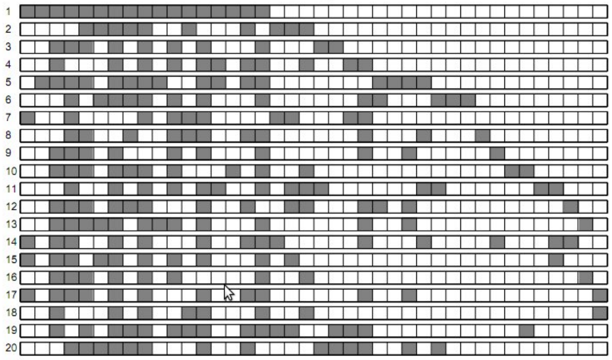

Figure 1: A binary matrix generated by the Indian buffet process with $\alpha = 10$

3.2 The Beta Process and the Bernoulli Process

The beta process (BP), proposed by Hjort [12] for applications in survival analysis, is a particular kind of independent increment process, or Lévy process, which can be defined as follows:

Let Ω be a space and $\mathcal{B}$ be the set of all measurable subsets of Ω. A beta process $B \sim BP(c, B_0)$ is a distribution on positive random measures over Ω, where c is a positive function over Ω, called the *concentration function*, and B_0 is a fixed continuous measure on $(\Omega, \mathcal{B})$, called the *base measure*. When c is a constant, we call it the *concentration parameter*. Assume $B_0(\Omega) < \infty$, then for any infinitestimal set $d\omega \in \mathcal{B}$, B is a beta process if

$$B(d\omega) \sim Beta(cB_0(d\omega), c(1 - B_0(d\omega)). \tag{3}$$

Every Lévy process is characterized by its *Lévy measure* or *rate measure*. The Lévy measure of the beta process $BP(c, B_0)$ is given by

$$\nu(d\omega, dp) = c(\omega)p^{-1}(1-p)^{c(\omega)-1}dpB_0(d\omega). \tag{4}$$

To draw B from the beta process distribution, we first draw a set of points $(\omega_i, p_i) \in \Omega \times [0, 1]$ from a Poisson process with rate measure ν, where Ω represents a space of atoms and $[0, 1]$ is the space of weights associated with these atoms. Then we define

$$B = \sum_{i=1}^{\infty} p_i \delta_{\omega_i}. \tag{5}$$

This discrete random measure is such that for any measurable set $S \in \Omega$, we have $B(S) = \sum_{i:\omega_i \in S} p_i$.

The beta process provides an infinite collection of probabilities that can be used to parametize the Bernoulli process. Let $B = \sum_{k=1}^{\infty} p_k \delta_{\omega_k}$ be a draw from a beta process. Let the row vector q_i be infinite and binary with the k^{th} value, q_{ik}, generated by

$$q_{ik} \sim Bernoulli(p_k). \tag{6}$$

Then the new measure $X_i = \sum_{k=1}^{\infty} q_{ik} \delta_{\omega_k}$ is a draw from the Bernoulli process $BeP(B)$.

We can link the beta process $B \sim BP(c, B_0)$ and N Bernoulli process iid draws, $X_i \sim BeP(B)$ for $n \in \{1, \ldots, N\}$, to generate a random feature matrix Z. To do so, we reorder the atoms in the beta process so that the first K are locations where some Bernoulli process in $\{X_i\}_{n=1}^{N}$ has a non-zero point mass. We further assume that c is a constant and B_0 is continuous with finite total mass $B_0(\Omega) = \gamma$. It turns out that the distribution of the matrix Z, with the atoms in the beta process draw integrated out, is the same as the distribution in (1). In other words, the beta process is the de Finetti mixing distribution underlying the Indian buffet process [23]. We refer to this overall procedure as the beta-Bernoulli process, denoted by $Z \sim BP\text{-}BeP(N, \gamma, c)$. Figure 2 illustrates draws from $BP\text{-}BeP(N, \gamma, c)$ for $N = 20$ and for several values of γ and c [23].

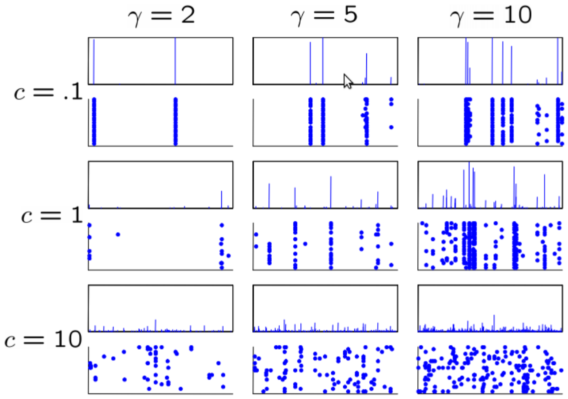

Figure 2: Draws from a beta process with several values of concentrationtion c and uniform base measure with mass γ For each draw, 20 samples are shown from the corresponding Bernoulli process, one per line

3.3 Infinite Sparse Factor Models

We first briefly review Principal Components Analysis (PCA), Independent Components Analysis (ICA), and Factor Analysis (FA). Let $y_n \in \Re^D$ be one observed data point and $x_n \in \Re^K$ be a vector of hidden components (both of them are row vectors). Based on these models, we can explain y_n in terms of a linear superposition of x_n as

$$y_n = x_n \mathbf{G} + \epsilon_n, \tag{7}$$

where $\mathbf{G}$ is the factor loading matrix and ϵ_n is a noise vector. PCA and FA assume that the latent factors come from normal priors. The difference between the two models is that in PCA, the noise is assumed to be isotropic, while in FA, the noise is assumed to be diagonal. In ICA, the latent factors are assumed to be heavy-tailed,

Knowles and Ghahramani [13, 14] propose a Bayesian nonparametric extension of these models by introducing the IBP into the models. First, the IBP introduces sparsity in the models. Sparsity is desirable in many applications mainly because it improves predictive performance and makes the models more interpretable. Second, the IBP enables the models to automatically decide how many hidden components to use. We illustrate the graphical model of the model in Figure 3.

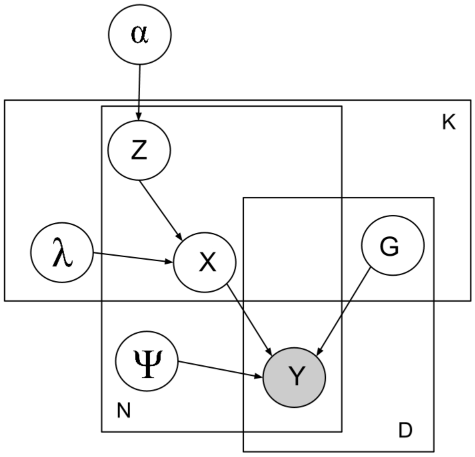

Figure 3: The graphical model of the infinite sparse factor analysis

Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{E}$ be concatenated matrices of x_n, y_n, z_n and ϵ_n, respectively, and $\odot$ denote an element-wise multiplication. The model places the IBP prior on $\mathbf{Z}$ both to allow an infinite number of columns and to induce sparsity of the latent factors, $\mathbf{X}$. That is,

$$\mathbf{Y} = (\mathbf{X} \odot \mathbf{Z})\mathbf{G} + \mathbf{E}, \tag{8}$$

where

$$\begin{array}{cc}
\epsilon_n \sim N(0, \Psi_n) & \Psi_n \sim IG(a, b) \\
Z \sim IBP(\alpha, 1) & \alpha \sim Gamma(e, f) \\
x_{nk} \sim Z_{nk} N(0, \lambda_k^{-1}) + (1 - Z_{nk})\delta_0(x_{nk}) & \lambda_k \sim Gamma(c, d) \\
g_{kd} \sim N(0, 1), &
\end{array}$$

and δ_0 is a delta function at 0.

Recently, Paisley et al. [19] propose the beta process factor analysis. In this model, they use the beta-Bernoulli process approximation as a prior to Z:

$$\begin{aligned}
z_{nk} &\sim Bernoulli(p_k) \\
p_k &\sim Beta(a/K, b(K-1)/K)
\end{aligned}$$

for $i = 1, \ldots, N$ and $k = 1, \ldots, K$. The rest of the model follows the generative model of the IBP infinite factor analysis.

Our model for learning image attributes will be based on this infinite sparse factor analysis model. We describe experimental setup, inference algorithm, results, and evaluation in the next sections.

4 Experimental Setup

4.1 Datasets

Currently, our factor analysis models do not explicitely capture spatial information among image pixels. Therefore, it is reasonble for us to select datasets whose images are roughly aligned. Given this information about the datasets, we can compute feature descriptions directly from images without the need to detect interest points. We perform inference algorithms on two datasets. One is the Caltech101 dataset [7], which consists of pictures of objects belonging to 101 categories. There are about 40 to 800 images per category, but most categories have about 50 images. In our experiments, we are focusing on the following seven categories: face ($N = 435$), chair ($N = 61$), lamp ($N = 61$), emu ($N = 53$), flamingo ($N = 67$), windsor chair ($N = 53$), and background ($N = 468$).

The other dataset comes from Sudderth et al. [22]. It consists of images from 16 categories, which fall into three groups: seven animal faces, five animal profiles, and four wheeled vehicles. Figure 3 [22] shows sample images from each category. The dataset also includes images from two additional categories: cannon and human face, as well as the background category. For convenience, we will refer to this dataset as the sixteen-category dataset.

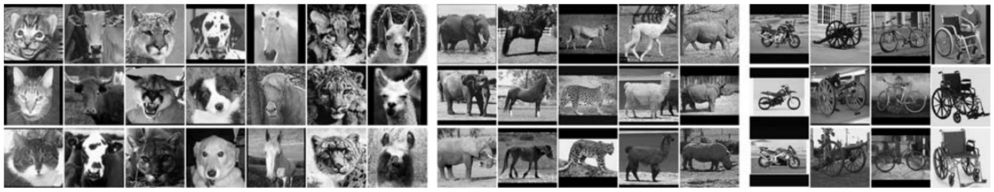

Figure 4: The sixteen-category dataset [22], which consists of a total of $1,885$ images with at least 50 images per category. Images in the dataset come from searches, the Caltech 101 and Weizmann Institute [3] datasets.

4.2 Feature Description

One of the most important steps in any vision task is selecting image representations. Using pixel intensities to represent an image has disadvantages because image intensities are sensitive to small shifts, rotations, and changes in illumination. Instead, our model exploits Histogram of Oriented Gradients (HOG) descriptors [4], which have been widely used in computer vision for the purpose of object detection. They have also been successfully applied to attribute recognition tasks [6, 5]. We expect HOG descriptors to be good at describing parts and textures of an object.

HOG descriptors describe an image with the distribution of intensity gradients or edge directions. To achieve the implementation of these descriptors, we divide the image into regions and then compute histogram of gradient directions for each region. We use the specific implementation from Felzenszwalb et al. [8] with bin size set to 8. In the end, each image in Caltech101 can be represented by a 693-dimensional HOG feature vector, while each image in the sixteen-category dataset can be represented by a $1,089$-dimensional vector.

5 Learning Image Attributes

5.1 Gibbs sampling

Given the observations Y (i.e. HOG features of images), we would like to infer the binary matrix Z, which specifies which image contains which attributes, the strengths of images' attributes X, the mixing matrix G, which defines what each attribute looks like, and all of the hyperparameters. We used a collapsed Gibbs sampler [14] to do the inference, but with Metropolis-Hastings steps for sampling new features. At each step, we draw samples from the marginal distribution of the model parameters given the data by successively sampling the conditional distributions of each parameter given all others. We will use the notation $-$ to denote the rest of the variables that are not explicitely conditioned upon in the current state of the Markov chain. The r^{th} row and the c^{th} column of a matrix A will be denoted by $A_{r:}$ and $A_{:c}$, respectively.

First, we note that our likelihood function is

$$P(\mathbf{Y}|\mathbf{X},\mathbf{G},\Psi) = \prod_{n=1}^{N} \frac{1}{(2\pi)^{D/2}|\Psi|^{1/2}} \times \exp(-\frac{1}{2}(\mathbf{y}_n - \mathbf{x}_n\mathbf{G})^T\Psi^{-1}(\mathbf{y}_n - \mathbf{x}_n\mathbf{G})). \tag{9}$$

Then we will specify the update formulas for each step in the sampling algorithm, as well as a method for adding new features.

Update for Z and X We first specify the ratio of likelihoods and then integrate out the element x_{nk} to obtain

$$\frac{P(\mathbf{Y}|\mathbf{Z}_{nk}=1,-)}{P(\mathbf{Y}|\mathbf{Z}_{nk}=0,-)} = \frac{\int P(\mathbf{Y}|x_{nk},-)N(x_{nk};0,\lambda_k^{-1})dx_{nk}}{P(\mathbf{Y}|x_{nk}=0,-)} = \sqrt{\frac{\lambda_k}{\lambda}}\exp(\frac{1}{2}\lambda\mu^2), \tag{10}$$

where $\lambda = \Psi_n^{-1}G_{k:}^T G_{k:} + \lambda_k$ and $\mu = \frac{\psi_n^{-1}}{\lambda}G_{k:}^T\hat{E}_n$ with the matrix of residuals $\hat{\mathbf{E}} = \mathbf{Y} - \mathbf{XG}$ evaluated with $x_{nk} = 0$.

From the exchangeability property of the IBP, we can imagine that the n^{th} image was the last to observed, so that the ratio of the priors is

$$\frac{P(Z_{nk}=1|-)}{P(Z_{nk}=0|-)} = \frac{m_{-n,k}}{N-1-m_{-n,k}}, \tag{11}$$

where $m_{-n,k} = \sum_{s\neq n} z_{s,k}$.

Multiplying equations (10) and (11) gives the expression for the ratio of the posterior probabilities for z_{nk}, which will be used to sample. If z_{nk} is set to 1, we sample $x_{nk}|- \sim N(\mu, \lambda^{-1})$.

Adding new features Let κ_n be the number of columns of $\mathbf{Z}$ which contain 1 only in row n. New features are proposed by sampling κ_n with a Metropolis-Hastings step. We integrate out $\mathbf{G}$ and include $\mathbf{x}$ as part of our proposal. That is, the proposal is $\xi = \{\kappa_n, \mathbf{x}\}$, and a move $\xi \to \xi^*$ is proposed with probability $J(\xi^*|\xi)$. The simplest proposal would be to use the prior on ξ^*

$$J(\xi) = P(\kappa_n|\alpha)P(\mathbf{x}|\kappa_n,\lambda_k) = Poisson(\kappa_n;\gamma)N(\mathbf{x};0,\lambda_k^{-1}), \tag{12}$$

where $\gamma = \frac{\alpha}{N-1}$.

The rate constant γ tends to be very small, resulting new features being rarely proposed. This problem can be fixed by introducing two new tunable parameters, π and ω, so the proposal distribution for κ_n becomes

$$J(\kappa_n) = (1-\pi)Poisson(\kappa_n;\omega\gamma) + \pi\mathbf{1}(\kappa_n = 1) \tag{13}$$

The rest of the updates are straightforward.

Update for G

$$P(\mathbf{g}_d|-) \propto P(\mathbf{y}_d|\mathbf{g}_d,-)P(\mathbf{g}_d) = N(\mathbf{g}_d;\mu_d,\mathbf{\Lambda}), \tag{14}$$

where $\mathbf{\Lambda} = \mathbf{X}^T\Psi^{-1}\mathbf{X} + I$ and $\mu_d = \mathbf{\Lambda}^{-1}\mathbf{X}^T\Psi^{-1}\mathbf{y}_n$.

Update for λ

$$P(\lambda_k|G) \sim Gamma(c + \frac{m_k}{2}, d + \sum_n X_{nk}^2). \tag{15}$$

Update for Ψ

$$P(\Psi_n^{-1}|-) \sim Gamma(a + \frac{N}{2}, b + \sum_n E_{nk}^2), \tag{16}$$

where $\hat{\mathbf{E}} = \mathbf{Y} - \mathbf{XG}$.

Update for α

$$P(\alpha|\mathbf{Z}) \propto P(\mathbf{Z}|\alpha)P(\alpha) = Gamma(\alpha; K_+ + e, f + H_N), \tag{17}$$

5.2 Results and Discussion

There are a total of 83 attributes for face images. However, each image only uses a few number of attributes. We display the top attributes for face images in Figure 5. In Figure 6, we show the matrices $X \odot Z$ for Face, Chair, and Lamp categories. Each matrix represents attribute strengths among images (darker means stronger). The darkest column of the face matrix corresponds to the first attribute in Figure 5.

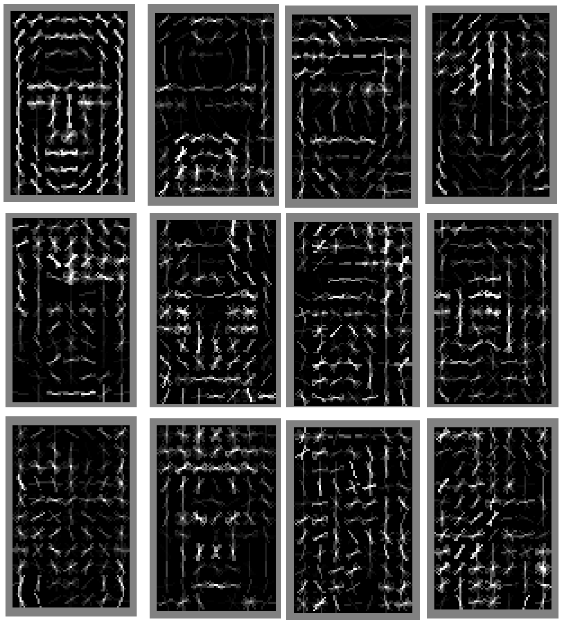

Figure 5: Top face attributes learned by the collapsed Gibbs sampling. We found the top attributes by summing up over rows of $X \odot Z$, and picked the 12 columns with highest values

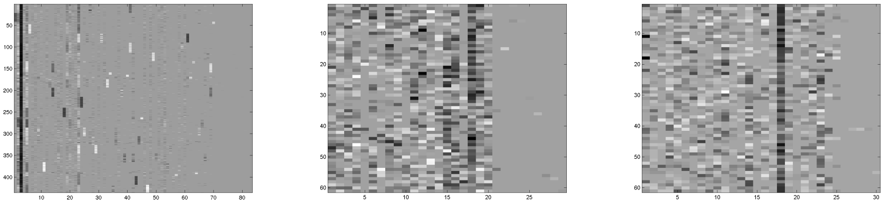

Figure 6: The matrices $X \odot Z$ learned by Gibbs for Face (left), Chair (middle), Lamp (right). Rows represent images and columns represent attributes

In Figure 7, we select some attributes from each category and show the images with strongest signal of those attributes. Images with the same attributes seem to have noticable visual similarities. We also see that the learned attributes are affected significantly by the object positions and orientations, as in the case of chairs. Furthermore, some attributes are reserved for one or a few images, which demonstrates our model's ability to capture irregularities among images.

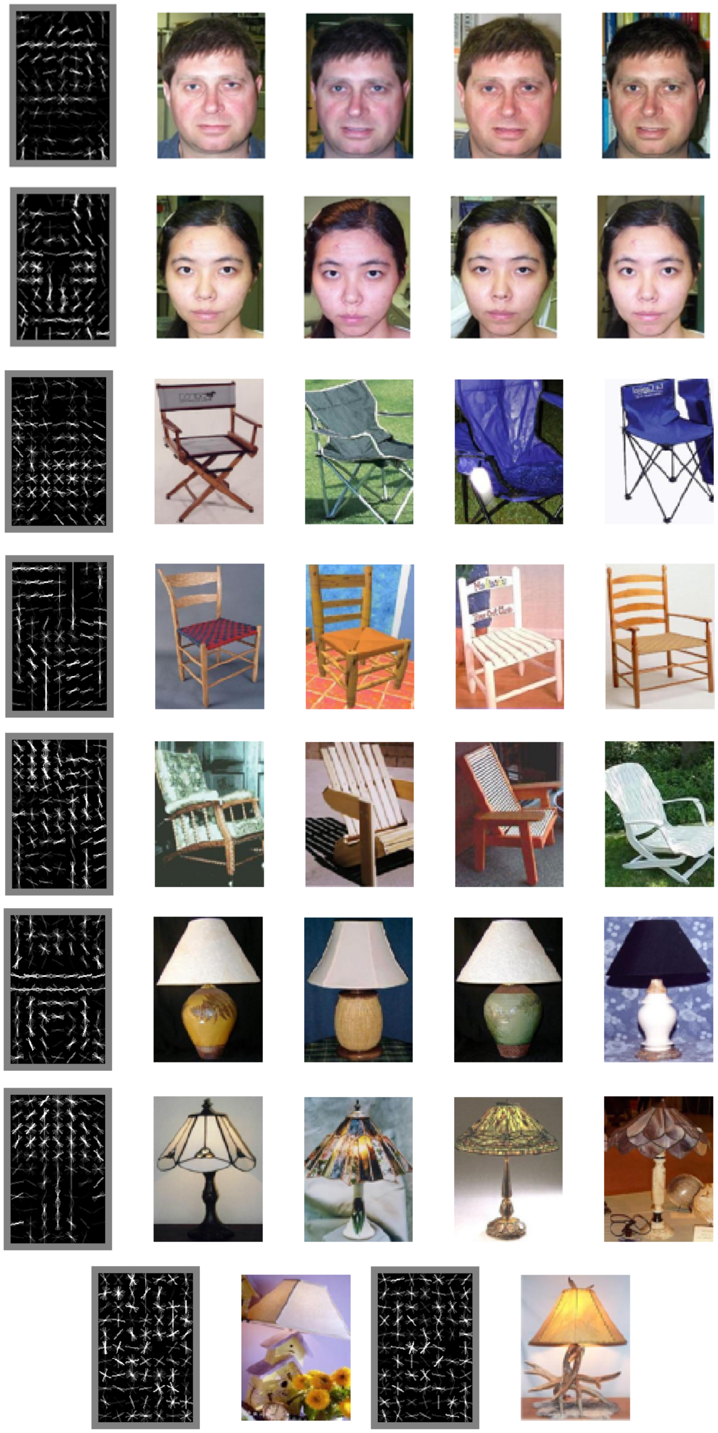

Figure 7: Attributes and images that are strongly related to those attributes (strengths are sorted in a decreasing order from left to right)

Next, in Figure 8, we show images and their top attributes. Face images share the strongest attribute, which captures the basic outline of all faces. In addition, there are attributes for different hairstyles

Table 1: Databases of images from the sixteen-category dataset

Database	Image Categories
I	(Side-viewed) Elephant, Horse, Leopard, Rhino
II	Cat face, Dog face, Leopard face, Cow head
III	(Side-viewed) Bike, Motorbike, Cannon, Car

and beard. Our model can also capture different face orientations of the same person. For example, the first two images share the top two attributes but the first image does not have the third attribute of the second image, due to the difference in face orientations.

In general attributes are not spatially localized as we might have hoped, mainly because there are some limitations in our model. For instance, there is no prior bias that they should be spatially localized. Moreover, images are not precisely aligned and there are no latent variables in the model that are responsible for attribute positions, so attributes need to model global translations. Lastly, the basic IBP cannot model correlations in feature usage.

6 Evaluation

We evaluated our model on the following vision tasks and compared them to simple baselines.

6.1 Image Retrieval

We begin by evaluating our model on an image retrieval task: given a single image, a retrieval system must provide a ranking of images in the database. Since we do not have relevance information from the users, we consider two images as relevant if and only if they are in the same category. We built three databases of images from the sixteen-category dataset. Each database consists of 100 images from 4 similar categories, shown in Table 1. We seperated 20 images from each category and used them as queries to our retrieval system later on.

The score of an image in the database is the dot product of that image's feature vector and the test image's feature vector (i.e., the linear kernel). In our baseline method, we used a 1089-dimensional vector of HOG features to represent an image. In our attributed-based method, we represented each image with a vector of attribute weights. First, we ran the infinite sparse factor analysis model on images from each category so as to find their attributes. Given all those attributes, we used Gibbs sampling to derive an attribute weight vector for each image in the database. Figure 9 shows attributed-based representations of images in a database, where images from different categories seem to have different attribute assignments. To infer the attribute weight vector of an image query, we ran the Gibbs sampler again, given attribute definition matrix, HOG features of images in the database along with their attribute weight vectors.

We submitted 80 image queries (20 per category) for each database. We compared the two rankings returned by our baseline and attributed-based models. Table 2 shows the mean average precision of the rankings for each image category. We can see that our attributed-based features can significantly improve retrieval effectiveness in all cases. Overall, we see a 30.11% improvement in the ranking performance.

6.2 Transfer Learning

Transfer learning can be viewed as generalization of knowledge; the knowledge learned in one situation might be useful in another. In the domain of object detection, transfer learning is especially important because the object space is extremely large, and it is often the case that we are in a situation where we see things we have not seen before. In this subsection, we test the ability of our attributed-based models to perform transfer learning. In particular, given a large number of images in one category as well as background images, we would like to improve the detection rate of objects of similar type. In the transfer learning paradigm, we only have a few of examples of the target class.

Table 2: Mean average precision (MAP) of rankings based on HOG features (HOG) and attribute-based features (ATT)

Category	**HOG**	**ATT**
Elephant	0.591	**0.872**
Horse	0.377	**0.686**
Leopard	0.577	**0.870**
Rhino	0.408	**0.713**
Cat face	0.588	**0.735**
Dog face	0.592	**0.763**
Leopard face	0.584	**0.735**
Cow head	0.795	**0.885**
Motorbike	0.975	**1.000**
Bike	0.768	**0.962**
Car	0.940	**0.999**
Cannon	0.256	**0.481**
All	0.621	**0.808**

We selected two pairs of categories from the Caltech101 dataset: Emu and Flamingo (different kinds of birds), as well as Windsor chair and Chair (one is a subset of the other). We learned an attribute definition matrix from positive (source images) and negative (background images) examples seperately. Given the attributes of the two, we inferred the attribute weight vectors of target images using the Gibbs samplers. We then trained SVM classifiers with the RBF kernel on $2-5$ examples of target images and tested on the held-out 30 images. There are three choices of image representations: HOG features (our baseline), attribute weight features, and a concatenation of the two. We report the average area under ROC curves over 15 trials in Figure 10. As shown, our attribute-based features give a higher detection accuracy in both cases.

The result suggests that attributes learned from images in one class are generic enough to be transferred to the learning of similar images in the other class. However, it is worth mentioning that the improvements in this transfer learning task are not always seen for many pairs of categories in the Caltech101 dataset. We suspect that this could be due to several reasons. For example, since all images in one category are roughly aligned, a few labeled images in the target category are enough to train an accurate detector, and thus the attributes do not really provide any additional information. In addition, we find out that, in some cases, attributes learned from the source images are too specific (such as fixed at some specific locations). These attributes cannot be transferred to the target images very well, even though they appear somewhere in the target images. Again, this could potentially be caused by the source images being roughly aligned. Finally, we discover that the detection accuracies are quite sensitive to the choices of attribute weight vectors inferred by the samplers. In the scenarios where we train our detectors using only a few examples, a wrong choice of attribute weight vectors can largely degrade the detection performance.

7 Conclusion and Future Work

We have proposed a nonparametric approach to the unsupervised learning of image attributes using the infinite sparse factor analysis. The Indian buffet process provides a Beyasian prior that introduces sparsity to our model. Also, the number of attributes does not have to be specified in advance due to the nonparametric property of the IBP. Experimental results show that attributes learned by our model are meaningful.

For future work, we plan to investigate into other inference algorithms. Moreover, the fact that our attributes are not spatially localized motivates us to model attribute positions, orientations, and correlations. We believe that doing so will make it far easier for us to perform transfer learning tasks. The transformed Indian buffet process introduced by Austerweil and Griffiths [1] seems to provide a good framework for such extension.

References

[1] J. L. Austerweil and T. L. Griffiths. Learning invariant features using the transformed Indian buffet process. In *NIPS*, pages 82–90, 2010.

[2] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *ECCV*, pages 663–676, 2010.

[3] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV*, pages 109–124, 2002.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, INRIA Rhône-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334, June 2005.

[5] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. In *CVPR*, pages 2352–2359. IEEE, 2010.

[6] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, pages 1778–1785, 2009.

[7] 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*, 2004.

[8] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Discriminatively trained deformable part models, release 4. http://people.cs.uchicago.edu/ pff/latent-release4/.

[9] R. Feris, B. Siddiquie, Y. Zhai, J. Petterson, L. Brown, and S. Pankanti. Attribute-based vehicle search in crowded surveillance videos. In *ACM ICMR*, pages 18:1–18:8, 2011.

[10] V. Ferrari and A. Zisserman. Learning visual attributes. In *NIPS*. MIT Press, 2007.

[11] Z. Ghahramani, T.L. Griffiths, and P. Sollich. Bayesian nonparametric latent feature models. *Bayesian Statistics*, 8:201–226, 2007.

[12] N. L. Hjort. Nonparametric Bayes estimators based on Beta processes in models for life history data. *Annals of Statistics*, 18:1259–1294, 1990.

[13] D. Knowles and Z. Ghahramani. Infinite sparse factor analysis and infinite independent components analysis. In *ICA*, pages 381–388, 2007.

[14] M. Knowles and Z. Ghahramani. Nonparametric bayesian sparse factor models with application to gene expression modeling. *The Annals of Applied Statistics*, 5:1534–1552, 2011.

[15] N. Kumar, P. N. Belhumeur, and S. K. Nayar. FaceTracer: A Search Engine for Large Collections of Images with Faces. In *ECCV*, pages 340–353, 2008.

[16] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372. IEEE, 2009.

[17] C.H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958, 2009.

[18] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[19] J. Paisley and L. Carin. Nonparametric factor analysis with beta process priors. In *ICML*, pages 777–784, 2009.

[20] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, pages 503–510, 2011.

[21] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where – and why? semantic relatedness for knowledge transfer. In *CVPR*, 2010.

[22] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky. Describing visual scenes using transformed objects and parts. *Int. J. Comput. Vision*, 77:291–330, May 2008.

[23] R. Thibaux and M. I. Jordan. Hierarchical beta processes and the indian buffet process. *Journal of Machine Learning Research - Proceedings Track*, 2:564–571, 2007.

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

[25] D. Vaquero, R. Feris, D. Tran, L. Brown, A. Hampapur, and M. Turk. Attribute-based people search in surveillance environments. In *WACV*, 2009.

[26] G. Wang and D. Forsyth. Joint learning of visual attributes, object classes and visual saliency. In *ICCV*, pages 537–544, 2009.

[27] J. Wang, K. Markert, and M. Everingham. Learning models for object recognition from natural language descriptions. In *BMVC*, 2009.

[28] X. Yu and Y. Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In *ECCV*, pages 127–140, 2010.

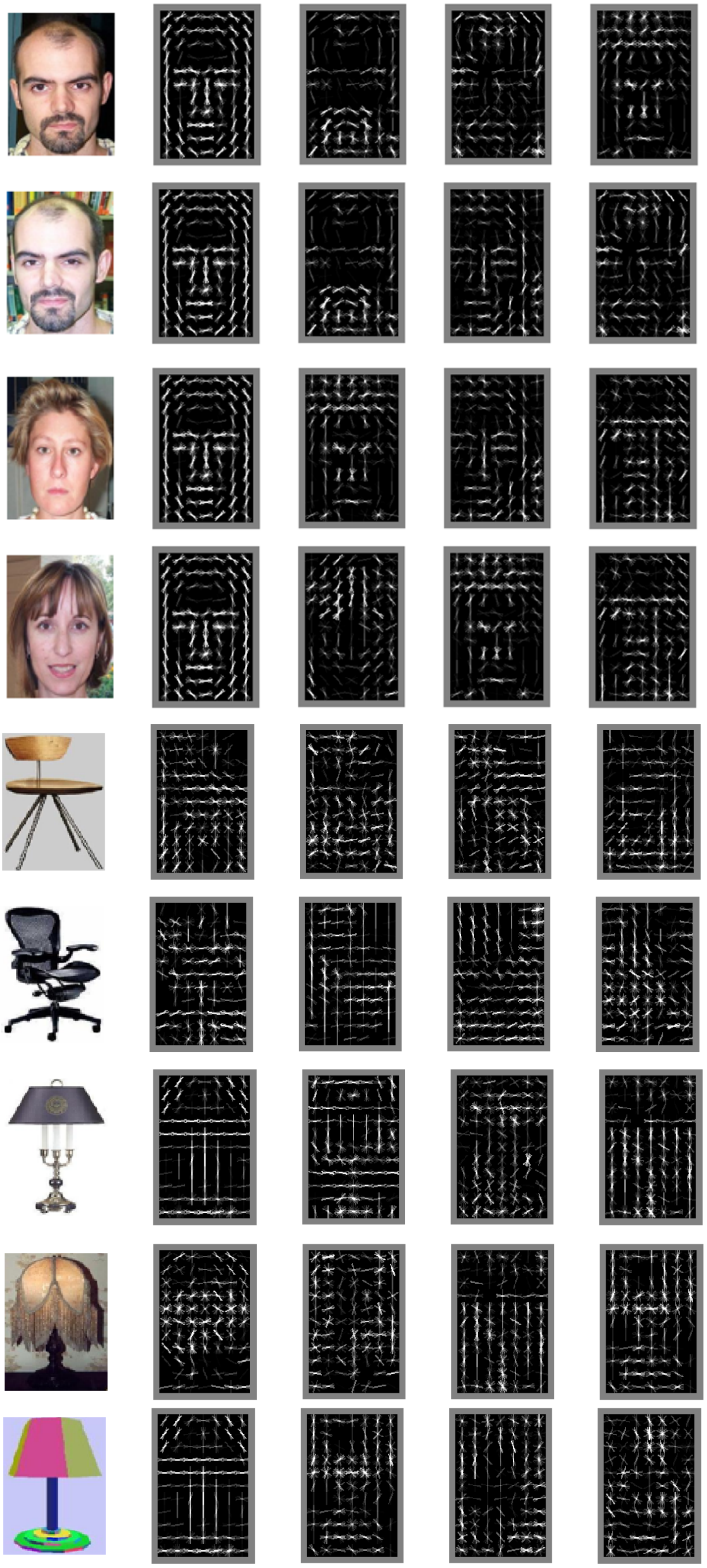

Figure 8: Images and attributes that are strongly related to those images (strengths are sorted in a decreasing order from left to right)

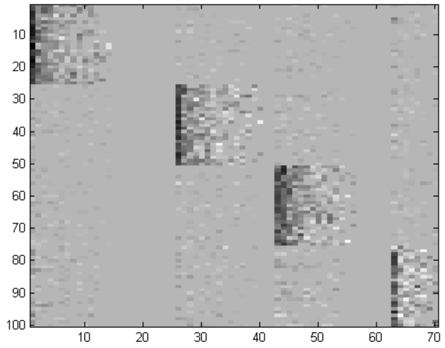

Figure 9: Attributed-based representations of images in the database

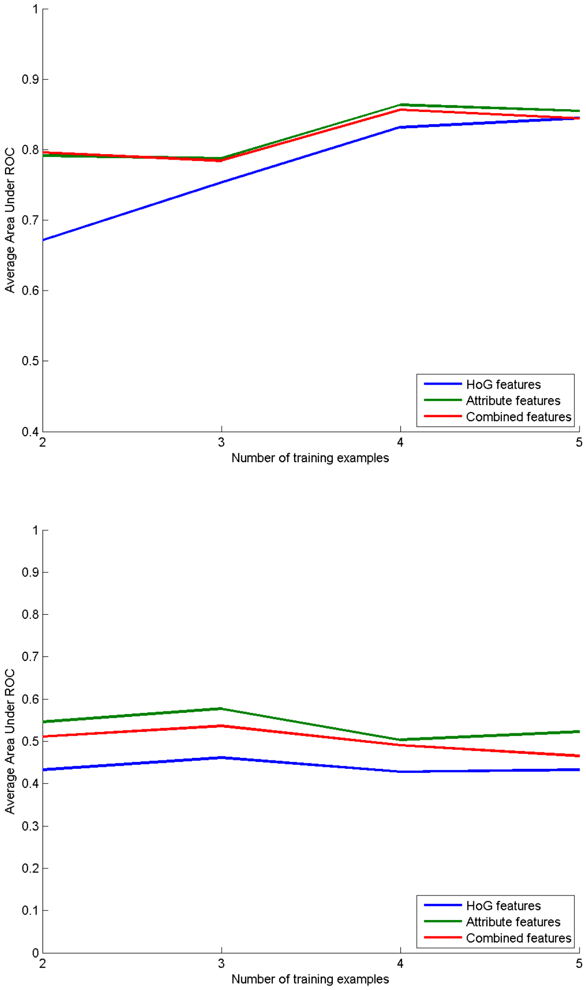

Figure 10: The average area under ROC curves in the tranfer learning task: Emu to Flamingo (top) and Windsor chair to Chair (bottom). The x-axis is the number of training examples per class