Generative versus discriminative methods for object recognition

Conference Paper *in* Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition · July 2005

DOI: 10.1109/CVPR.2005.167 · Source: IEEE Xplore

Ilkay Ulusoy
Middle East Technical University

Generative Versus Discriminative Methods for Object Recognition

Ilkay Ulusoy
METU, Electrical and Electronics Eng. Department, 06531 Ankara Turkey
ilkay@metu.edu.tr

Christopher M. Bishop
Microsoft Research, 7 J J Thomson Avenue Cambridge, CB3 0FB, U.K.
cmbishop@microsoft.com

Abstract

Many approaches to object recognition are founded on probability theory, and can be broadly characterized as either generative or discriminative according to whether or not the distribution of the image features is modelled. Generative and discriminative methods have very different characteristics, as well as complementary strengths and weaknesses. In this paper we introduce new generative and discriminative models for object detection and classification based on weakly labelled training data. We use these models to illustrate the relative merits of the two approaches in the context of a data set of widely varying images of non-rigid objects (animals). Our results support the assertion that neither approach alone will be sufficient for large scale object recognition, and we discuss techniques for combining them.

1. Introduction

Object recognition is currently one of the most actively researched areas of computer vision. Increasingly, it is being approached using machine learning techniques based on probability theory. While many models have been proposed, there has been little attempt at a systematic characterization of the different approaches. In this paper we show that it is useful to categorize them as either generative or discriminative. To understand the distinction consider a scenario in which an image described by a vector $\mathbf{X}$ (which might be raw pixel intensities, or some set of features extracted from the image) is to be assigned to one of K classes $k = 1, \ldots, K$. From basic decision theory [2] we know that the most complete characterization of the solution is expressed in terms of the set of posterior probabilities $p(k|\mathbf{X})$. Once we know these probabilities it is straightforward to assign the image $\mathbf{X}$ to a particular class to minimize the expected loss (for instance, if we wish to minimize the number of misclassifications we assign $\mathbf{X}$ to the class having the largest posterior probability).

In a discriminative approach we introduce a parametric model for the posterior probabilities, and infer the values of the parameters from a set of labelled training data. This may be done by making point estimates of the parameters using maximum likelihood, or by computing distributions over the parameters in a Bayesian setting (for instance by using variational inference).

By contrast, in a generative approach we model the joint distribution $p(k, \mathbf{X})$ of images and labels. This can be done, for instance, by learning the class prior probabilities $p(k)$ and the class-conditional densities $p(\mathbf{X}|k)$ separately. The required posterior probabilities are then obtained using Bayes' theorem

$$p(k|\mathbf{X}) = \frac{p(\mathbf{X}|k)p(k)}{\sum_j p(\mathbf{X}|j)p(j)} \tag{1}$$

where the sum in the denominator is taken over all classes.

Compared with discriminative approaches, generative models typically have the following advantages:

1. They can handle missing data or partially labelled data, and can augment small quantities of expensive labelled data with large quantities of cheap unlabelled data.

2. A new class $K + 1$ can be added incrementally by learning its class-conditional density $p(\mathbf{X}|K + 1)$ independently of all the previous classes.

3. Generative models can readily handle compositionality (e.g. faces with glasses and/or hats, and/or moustaches) whereas standard discriminative models need to see all combinations of possibilities during training.

By contrast, discriminative models generally offer the following advantages:

0-7695-2372-2/05/$20.00 (c) 2005 IEEE

1. The flexibility of the model is used in regions of input space where the posterior probabilities differ significantly from 0 or 1, whereas generative approaches model details of the distribution of $\mathbf{X}$ which may be irrelevant for determining the posterior probabilities.

2. Discriminative models are typically very fast at making predictions for new (test) data points, while generative models often require iterative solution.

3. Other things being equal it would be expected that discriminative methods would have better predictive performance since they are trained to predict the class label rather than the joint distribution of input vectors and targets.

A key issue in object recognition is the need for predictions to be invariant to a wide variety of transformations of the input image due to translations and rotations of the object in 3D space, changes in viewing direction and distance, variations in the intensity and nature of the illumination, and non-rigid transformations of the object itself. Another key issue is to recognize the object even if it is occluded. In most of the cases, features obtained from image patches are used as a solution to these problems. Informative features selected using some information criterion versus generic features were compared in [11] and although the informative features used were shown to be superior to generic features when used with a simple classification method, they are not invariant to scale and orientation. By contrast, generic interest point operators such as saliency [6], DoG [7] and Harris-Laplace [9] detectors are repeatable in the sense that they are invariant to location, scale and orientation, and some are also affine invariant [7, 9] to some extent. For the purposes of this paper we shall consider the use of invariant features obtained from local regions of the image. However, in Section 6 we shall consider the relative merits of generative and discriminative approaches in the context of learning invariant features from data.

In the hierarchy of object recognition problems, one upper level is the localization of the object in the views. Fergus et al. [5] learn jointly the appearances and relative locations of a small set of parts whose potential locations are determined by a saliency detector [6]. Since their algorithm is very complex, the number of parts has to be kept small. Based on [3], which performs multiclass object recognition but cannot detect object in the view, [4] tried to find out informative features, which are expected to be the object features, based on information criteria such as likelihood ratio and mutual information. However, in this supervised approach, hundreds of images were hand segmented in order to train support vector machine and Gaussian mixture models (GMMs) for foreground (i.e. object) background classification. Finally, Xie and Perez [12] extended the GMM based approach of [4] to a semi-supervised case inspired from [5]. A multi-modal GMM was trained to model foreground and background features where some uncluttered images of foreground were used for the purpose of initialization. In both our discriminative and generative approaches, we explore not only the labelling of images according to the object categories present, but also the labelling of each feature (interest point) as a form of object localization.

2. Object Recognition

Much of our discussion of generative and discriminative models has wide applicability. In this paper, however, we focus on object recognition which has emerged as a ‘grand challenge’ for computer vision, with the longer term aim of being able to achieve near human levels of recognition for tens of thousands of object categories under a wide variety of conditions.

Our goal in this paper is not to find optimal features and representations for solving a specific object recognition task, but rather to fix on a particular, widely used, feature set and use this as the basis to compare alternative learning methodologies. We shall also fix on a specific data set, chosen for the wide variability of the objects in order to present a non-trivial classification problem. In particular, we consider the problem of detecting and distinguishing cows and sheep in natural images.

We therefore follow several recent approaches [7, 9] and use an interest point detector to focus attention on a small number of local patches in each image. This is followed by invariant feature extraction from a neighbourhood around each interest point. Specifically we use DoG interest point detectors, and at each interest point we extract a 128 dimensional SIFT feature vector [7]. Following [1] we concatenate the SIFT features with additional colour features comprising average and standard deviation of (R, G, B), (L, a, b) and $(r = R/(R+G+B), g = G/(R+G+B))$, which gives an overall 144 dimensional feature vector. The result of applying the DoG operator to a cow image is shown in Figure 1 where squares are centered at the interest points and width of the square show the scale of the interest point. The SIFT descriptors and colour features are obtained from these square patches.

In this paper we use $\mathbf{t}_n$ to denote the image label vector for image n with independent components $t_{nk} \in \{0, 1\}$ in which $k = 1, \ldots K$ labels the class. Each class can be present or absent independently in an image, and we make no distinction between foreground and background classes within the model itself. $\mathbf{X}_n$ denotes the observation for image n and this comprises as set of J_n patch vectors $\{\mathbf{x}_{nj}\}$ where $j = 1, \ldots, J_n$. Note that the number J_n of detected interest points will in general vary from image to image.

On a small-scale problem it is reasonable to segment and label the objects present in the training images. However, for large-scale object recognition involving thousands of categories this will not be feasible, and so instead it is necessary to employ training data which is at best 'weakly labelled'. Here we consider a training set in which each image is labelled only according to the presence or absence of each category of object (cows and sheep in our example).

Next we associate with each patch j in each image n a binary label $\tau_{njk} \in \{0,1\}$ denoting the class k of the patch. For the models developed in this paper we shall consider these labels to be mutually exclusive, so that $\sum_{k=1}^{K} \tau_{njk} = 1$, in other words each patch is assumed to be either cow, sheep or background. These components can be grouped together into vectors $\boldsymbol{\tau}_{nj}$. Note that this assumption is not essential, and other formulations could also be considered. If the values of these labels were available during training (corresponding to strongly labelled images) then the development of recognition models would be greatly simplified. For weakly labelled data, however, the $\{\boldsymbol{\tau}_{nj}\}$ labels are hidden (latent) variables, which of course makes the training problem much harder.

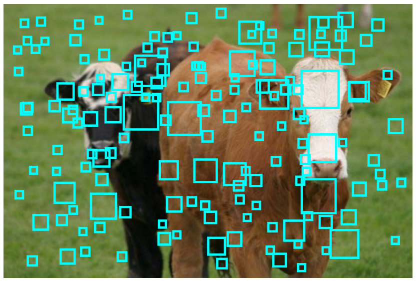

Figure 1. Difference of Gaussian interest points with their local regions. Note that interest points fall both on the objects of interest (the cows) and also on the background.

3. The Discriminative Model

We begin by introducing a discriminative model, which corresponds to the directed graph shown in Figure 2. Consider for a moment a particular image n (and omit the index n to keep the notation uncluttered). We build a parametric model $y_k(\mathbf{x}_j, \mathbf{w})$ for the probability that patch $\mathbf{x}_j$ belongs to class k. For example we might use a simple linear-softmax model with outputs

$$y_k(\mathbf{x}_j, \mathbf{w}) = \frac{\exp(\mathbf{w}_k^{\mathrm{T}} \mathbf{x}_j)}{\sum_l \exp(\mathbf{w}_l^{\mathrm{T}} \mathbf{x}_j)} \tag{2}$$

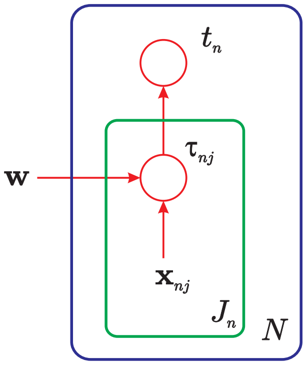

Figure 2. Graphical representation of the discriminative model for object recognition.

which satisfy $0 \leqslant y_k \leqslant 1$ and $\sum_k y_k = 1$. More generally we can use a multi-layer neural network, relevance vector machine, etc. The probability of a patch label $\boldsymbol{\tau}_j$ to be class k is then given directly in terms of the outputs $\{y_k\}$

$$p(\boldsymbol{\tau}_j | \mathbf{x}_j) = \prod_{k=1}^{K} y_k(\mathbf{x}_j, \mathbf{w})^{\tau_{jk}}. \tag{3}$$

Next we assume that if one, or more, of the patches carries the label for a particular class, then the whole image will. For instance, if there is at least one local patch in the image which is labelled 'cow' then the whole image will carry a 'cow' label (recall that an image can carry more than one class label at a time). Thus the conditional distribution of the image label, given the patch labels, is given by

$$\begin{aligned} p(\mathbf{t}|\boldsymbol{\tau}) &= \prod_{k=1}^{K} \left[1 - \prod_{j=1}^{J} [1 - \tau_{jk}] \right]^{t_k} \\ &\quad \left[\prod_{j=1}^{J} [1 - \tau_{jk}] \right]^{1-t_k}. \end{aligned} \tag{4}$$

In order to obtain the conditional distribution $p(\mathbf{t}|\mathbf{X})$ we have to marginalize over the latent patch labels. Although there are exponentially many terms in this sum, it can be performed analytically for our model to give

$$\begin{aligned} p(\mathbf{t}|\mathbf{X}) &= \sum_{\boldsymbol{\tau}} \left\{ p(\mathbf{t}|\boldsymbol{\tau}) \prod_{j=1}^{J} p(\boldsymbol{\tau}_j | \mathbf{x}_j) \right\} \\ &= \prod_{k=1}^{K} \left[1 - \prod_{j=1}^{J} [1 - y_k(\mathbf{x}_j, \mathbf{w})] \right]^{t_k} \\ &\quad \left[\prod_{j=1}^{J} [1 - y_k(\mathbf{x}_j, \mathbf{w})] \right]^{1-t_k}. \end{aligned} \tag{5}$$

This can be viewed as a softened (probabilistic) version of the 'OR' function as used in [8].

Given a training set of N images, which are assumed to be independent, we can construct the likelihood function from the product of such distributions, one for each data point. Taking the negative logarithm then gives the following error function

$$E(\mathbf{w}) = -\sum_{n=1}^{N}\sum_{k=1}^{K}\{t_{nk}\ln[1-Z_{nk}] + (1-t_{nk})\ln Z_{nk}\} \tag{6}$$

where we have defined

$$Z_{nk} = \prod_{j=1}^{J_n}[1-y_k(\mathbf{x}_{nj},\mathbf{w})]. \tag{7}$$

The parameter vector $\mathbf{w}$ can be determined by minimizing this error (which corresponds to maximizing the likelihood function) using a standard optimization algorithm such as scaled conjugate gradients [2]. More generally the likelihood function could be used as the basis of a Bayesian treatment, although we do not consider this here.

Once the optimal value $\mathbf{w}_{\mathrm{ML}}$ is found, the corresponding functions $y_k(\mathbf{x},\mathbf{w}_{\mathrm{ML}})$ for $k = 1,\ldots,K$ will give the posterior class probabilities for a new patch feature vector $\mathbf{x}$. Thus the model has learned to label the patches even though the training data contained only image labels. Note, however, that as a consequence of the noisy 'OR' assumption, the model only needs to label one foreground patch correctly in order to predict the image label. It will therefore learn to pick out a small number of highly discriminative foreground patches, and will classify the remaining foreground patches, as well as those falling on the background, as 'background' meaning non-discriminative for the foreground class. This will be illustrated in Section 5.

4. The Generative Model

Next we turn to a description of our generative model, whose graphical representation is shown in Figure 3. The structure of this model mirrors closely that of the discriminative model. In particular, the same class-label variables $\boldsymbol{\tau}_{nj}$ are associated with the patches in each image, and again these are unobserved and must be marginalized out in order to obtain maximum likelihood solutions.

In the discriminative model we represented the conditional distribution $p(\mathbf{t}|\mathbf{X})$ directly as a parametric model. By contrast in the generative approach we model $p(\mathbf{t},\mathbf{X})$, which we decompose into $p(\mathbf{t},\mathbf{X}) = p(\mathbf{X}|\mathbf{t})p(\mathbf{t})$ and then model the two factors separately. This decomposition would allow us, for instance, to employ large numbers of 'background' images (those containing no instances of the object classes) during training without concluding that the prior probability of objects is small.

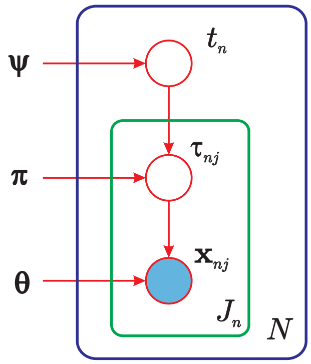

Figure 3. Graphical representation of the generative model for object recognition.

Again, we begin by considering a single image n. The prior $p(\mathbf{t})$ is specified in terms of K parameters ψ_k where $0 \leqslant \psi_k \leqslant 1$ and $k = 1,\ldots,K$, so that

$$p(\mathbf{t}) = \prod_{k=1}^{K}\psi_k^{t_k}(1-\psi_k)^{1-t_k}. \tag{8}$$

In general we do not need to learn these from the training data since the prior occurrences of different classes is more a property of the way the data was collected than of the real world frequencies. (Similarly in the discriminative model we will typically wish to correct for different priors between the training set and test data using Bayes' theorem.)

The remainder of the model is specified in terms of the conditional probabilities $p(\boldsymbol{\tau}|\mathbf{t})$ and $p(\mathbf{X}|\boldsymbol{\tau})$. The probability of generating a patch from a particular class is governed by a set of parameters π_k, one for each class, such that $\pi_k \geqslant 0$, constrained by the subset of classes actually present in the image. Thus

$$p(\boldsymbol{\tau}_j|\mathbf{t}) = \left(\sum_{l=1}^{K}t_l\pi_l\right)^{-1}\prod_{k=1}^{K}(t_k\pi_k)^{\tau_{jk}}. \tag{9}$$

Note that there is an overall undetermined scale to these parameters, which may be removed by fixing one of them, e.g. $\pi_1 = 1$.

For each class, the distribution of the patch feature vector $\mathbf{x}$ is governed by a separate mixture of Gaussians which we denote by

$$p(\mathbf{x}|\boldsymbol{\tau}_j) = \prod_{k=1}^{K}\phi_k(\mathbf{x}_j;\boldsymbol{\theta}_k)^{\tau_{jk}} \tag{10}$$

where $\boldsymbol{\theta}_k$ denotes the set of parameters (means, covariances and mixing coefficients) associated with this mixture model.

If we assume N independent images, and for image n we have J_n patches drawn independently, then the joint distribution of all random variables is

$$\prod_{n=1}^{N} p(\mathbf{t}_n) \prod_{j=1}^{J_n} p(\mathbf{x}_{nj}|\boldsymbol{\tau}_{nj}) p(\boldsymbol{\tau}_{nj}|\mathbf{t}_n). \tag{11}$$

Since we wish to maximize likelihood in the presence of latent variables, namely the $\{\boldsymbol{\tau}_{nj}\}$, we use the EM algorithm. The expected complete-data log likelihood is given by

$$\sum_{n=1}^{N}\sum_{j=1}^{J_n}\left\{\sum_{k=1}^{K}\langle\tau_{njk}\rangle \ln\left[t_{nk}\pi_k\phi_k(\mathbf{x}_{nj})\right] - \ln\left(\sum_{l=1}^{K} t_{nl}\pi_l\right)\right\}. \tag{12}$$

The expected values of τ_{nkj} are computed in the E-step using

$$\begin{aligned} \langle\tau_{njk}\rangle &= \sum_{\{\boldsymbol{\tau}_{nj}\}} \tau_{njk} p(\boldsymbol{\tau}_{nj}|\mathbf{x}_{nj}, \mathbf{t}_n) \\ &= \frac{t_{nk}\pi_k\phi_k(\mathbf{x}_{nj})}{\displaystyle\sum_{l=1}^{K} t_{nl}\pi_l\phi_l(\mathbf{x}_{nj})}. \end{aligned} \tag{13}$$

Notice that the first factor on the right hand side of (9) has cancelled in the evaluation of $\langle\tau_{njk}\rangle$.

For the M-step we first set the derivative with respect to one of the parameters π_k equal to zero (note there is no Lagrange multiplier since there is no summation constraint on the $\{\pi_k\}$) and re-arrange to give the following re-estimation equations

$$\pi_k = \left[\sum_{n=1}^{N} J_n t_{nk}\left(\sum_{l=1}^{K} t_{nl}\pi_l\right)^{-1}\right]^{-1}\sum_{n=1}^{N}\sum_{j=1}^{J_n}\langle\tau_{njk}\rangle. \tag{14}$$

Since these represent coupled equations we perform several (fast) iterations of these equations before proceeding with the next EM cycle (note that the sums over j can be precomputed).

Now consider the optimization with respect to the parameters $\boldsymbol{\theta}_k$ governing the distribution $\phi_k(\mathbf{x};\boldsymbol{\theta}_k)$. The dependence of the expected complete-data log likelihood on $\boldsymbol{\theta}_k$ takes the form

$$\sum_{n=1}^{N}\sum_{j=1}^{J_n}\langle\tau_{njk}\rangle \ln\phi_k(\mathbf{x}_{nj};\boldsymbol{\theta}_k) + \text{const.} \tag{15}$$

This is easily maximized for each class k separately using the EM algorithm (in an inner loop), since (15) simply represents a log likelihood function for a weighted data set in which patch (n, j) is weighted with $\langle\tau_{njk}\rangle$.

Specifically, we use a model in which $\phi_k(\mathbf{x};\boldsymbol{\theta}_k)$ is given by a Gaussian mixture distribution of the form

$$\phi_k(\mathbf{x};\boldsymbol{\theta}_k) = \sum_{m=1}^{M}\rho_{km}\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}). \tag{16}$$

The E-step is given by

$$\gamma_{njkm} = \frac{\rho_{km}\mathcal{N}(\mathbf{x}_{nj}|\boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km})}{\sum_{m'}\rho_{km'}\mathcal{N}(\mathbf{x}_{nj}|\boldsymbol{\mu}_{km'}, \boldsymbol{\Sigma}_{km'})} \tag{17}$$

while the M-step equations are weighted by the coefficients $\langle\tau_{njk}\rangle$ to give

$$\begin{aligned} \boldsymbol{\mu}_{km}^{\text{new}} &= \frac{\sum_n\sum_j\langle\tau_{njk}\rangle\gamma_{njkm}\mathbf{x}_{nj}}{\sum_n\sum_j\langle\tau_{njk}\rangle\gamma_{njkm}} \\ \boldsymbol{\Sigma}_{km}^{\text{new}} &= \frac{\sum_n\sum_j\langle\tau_{njk}\rangle\gamma_{njkm}(\mathbf{x}_{nj}-\boldsymbol{\mu}_{km}^{\text{new}})(\mathbf{x}_{nj}-\boldsymbol{\mu}_{km}^{\text{new}})^{\mathrm{T}}}{\sum_n\sum_j\langle\tau_{njk}\rangle\gamma_{njkm}} \\ \rho_{km}^{\text{new}} &= \frac{\sum_n\sum_j\langle\tau_{njk}\rangle\gamma_{njkm}}{\sum_n\sum_j\langle\tau_{njk}\rangle}. \end{aligned}$$

If one EM cycle is performed for each mixture model $\phi_k(\mathbf{x};\boldsymbol{\theta}_k)$ this is equivalent to a global EM algorithm for the whole model. However, it is also possible to perform several EM cycle for each mixture model $\phi_k(\mathbf{x};\boldsymbol{\theta}_k)$ within the outer EM algorithm. All of these variants yield valid EM algorithms in which the likelihood never decreases.

The incomplete-data log likelihood can be evaluated after each iteration to ensure that it is correctly increasing. It is given by

$$\sum_{n=1}^{N}\sum_{j=1}^{J_n}\left\{\ln\left(\sum_{k=1}^{K} t_{nk}\pi_k\phi_k(\mathbf{x}_{nj})\right) - \ln\left(\sum_{l=1}^{K} t_{nl}\pi_l\right)\right\}.$$

Note that, for a data set in which all $t_{nk} = 1$, the model simply reduces to fitting a flat mixture to all observations, and the standard EM is recovered as a special case of the above equations.

This model can be viewed as a generalization of that presented in [12] in which a parameter is learned for each mixture component representing the probability of that component being foreground. This parameter is then used to select the most informative N components in a similar approach to [4] and [11] where the number N is chosen heuristically. In our case, however, the probability of each feature belonging to one of the K classes is learned directly.

Inference in the generative model is more complicated than in the discriminative model. Given all patches $\mathbf{X} = \{\mathbf{x}_j\}$ from an image, the posterior probability of the label $\boldsymbol{\tau}_j$ for patch j can be found by marginalizing out all other hidden variables

$$p\left(\boldsymbol{\tau}_j|\mathbf{X}\right)=\sum_{\mathbf{t}}\sum_{\boldsymbol{\tau}/\boldsymbol{\tau}_j}p\left(\boldsymbol{\tau},\mathbf{X},\mathbf{t}\right)$$

$$=\sum_{\mathbf{t}}p\left(\mathbf{t}\right)\frac{1}{\left(\sum_{l=1}^{K}\pi_l t_l\right)^J}\prod_{k=1}^{K}\left(\pi_k t_k \phi_k\left(\mathbf{x}_j\right)\right)^{\tau_{jk}}$$

$$\prod_{i\neq j}\left[\sum_{k=1}^{K}\pi_k t_k \phi_k\left(\mathbf{x}_i\right)\right] \qquad (18)$$

where $\boldsymbol{\tau}=\{\boldsymbol{\tau}_j\}$ denotes the set of all patch labels, and $\boldsymbol{\tau}/\boldsymbol{\tau}_j$ denotes this set with $\boldsymbol{\tau}_j$ omitted. Note that the summation over all possible $\mathbf{t}$ values, which must be done explicitly, is computationally expensive.

Inference of image label needs almost as much computation as patch label inference where the posterior probability of image label $\mathbf{t}$ can be computed using

$$p\left(\mathbf{t}|\mathbf{X}\right)=\frac{p\left(\mathbf{X}|\mathbf{t}\right)p\left(\mathbf{t}\right)}{p\left(\mathbf{X}\right)} \qquad (19)$$

where $p(\mathbf{t})$ is computed from the data set, $p(\mathbf{X})$ is the normalization factor and $p\left(\mathbf{X}|\mathbf{t}\right)$ is calculated by integrating out patch labels

$$p\left(\mathbf{X}|\mathbf{t}\right) = \sum_{\boldsymbol{\tau}}\prod_{j=1}^{J}p\left(\mathbf{X},\boldsymbol{\tau}|\mathbf{t}\right)$$

$$= \prod_{j=1}^{J_n}\frac{\sum_{k=1}^{K}t_k\pi_k\phi_k\left(\mathbf{x}_j\right)}{\sum_{l=1}^{K}t_l\pi_l}. \qquad (20)$$

5. Results

In this study, we have used a test bed of weakly labelled images each containing either cows or sheep, in which the animals vary widely in terms of number, pose, size, colour and texture. There are 167 images in each class, and 10-fold cross-validation is used to measure performance. For the discriminative model we used a linear network of the form (2) with 144 inputs and 3 outputs (cow, sheep, background), and also two-layer non-linear networks having 50 hidden units with '$\tanh$' activation functions, and a quadratic regularizer with hyper-parameter 0.2. For the generative model we used a separate Gaussian mixture for cow, sheep and background, each of which has 10 components with diagonal covariance matrices.

Initial results with the generative model showed that with random initialization of the mixture model parameters it is incapable of learning a satisfactory solution. We conjectured that this is due to the problem of multiple local maxima in the likelihood function (a similar effect was found by [12]). To test this we used some segmented images for initialization purposes (but not for training). 30 cow and 30 sheep images were hand-segmented, and features belonging to each class were clustered using the K-means algorithm and the component centers of a class mixture model were assigned to cluster centers of the respective class. The mixing coefficients were set to the number of points in the corresponding cluster divided by the total number of points in that class. Also, covariance matrixes were computed using the data points assigned to the respective center.

The overall correct rate means and variances of object recognition, i.e. image labelling, are given in the first two rows of Table 1 for linear (L) and nonlinear (NL) discriminative (D) models and generative (G) model. When half of the data is used in training and the other half is used in testing, the results do not change significantly and are given in the last two rows of Table 1.

The performance of the generative model immediately after initialization and before running the EM algorithm gave an overall correct classification rate of 90% compared with 97% after training.

It is also interesting to investigate the extent to which the two models correctly label the individual patches. In order to make a comparison in terms of patch labelling we used 30 hand segmented images for each class. In Table 2 patch labelling scores for foreground (FG) and background (BG) for discriminative and generative models are given. Various thresholds are used on patch label probabilities in order to produce ROC curves for the generative model and the non-linear network version of the discriminative model, as shown in Figure 4. We also plot the ROC curve for the generative model when random initialization is performed to show the importance of initialization for such models.

As already noted, the discriminative model finds a small number of highly discriminative foreground patches, and labels all other patches as background, whereas the generative model must balance both foreground and background patches. Some examples of patch labelling for test images are given in Figure 5 for cow images and in Figure 6 for sheep images.

The hand segmented images were not used in training. But if they are intended to be used during training also, then it is very easy to insert these strongly labelled data in the generative model training. In the outer E step of the EM training of the generative model, expected patch labels are computed (13). When strongly labelled data are used then the known patch labels for the strongly labelled data are used instead of the expected values while expected patch labels for weakly labelled data are used as they are. When we used both weakly labelled and strongly labelled data in the generative model, the overall correct rate and patch labelling success increased slightly compared with the case when we used strongly labelled data only for initialization of the generative model.

There is a huge difference between discriminative and generative models in terms of speed. The generative model is more than 20 times slower than the discriminative model in training and more than 2×10^2 times slower in testing. Typical values for the duration of a single cycle and the total duration of training and testing are given, for a Matlab implementation, in Table 3.

Table 1. Overall correct rates (%).

	D-L	D-NL	G
mean	82.5	87.2	97
var	8.17	6.75	2.9
mean	83	84.5	93
var	6.9	3.7	0.52

Table 2. Patch labelling scores.

Class	D-BG	D-FG	G-BG	G-FG
Cow	99%	17%	82%	68%
Sheep	99%	5%	52%	82%

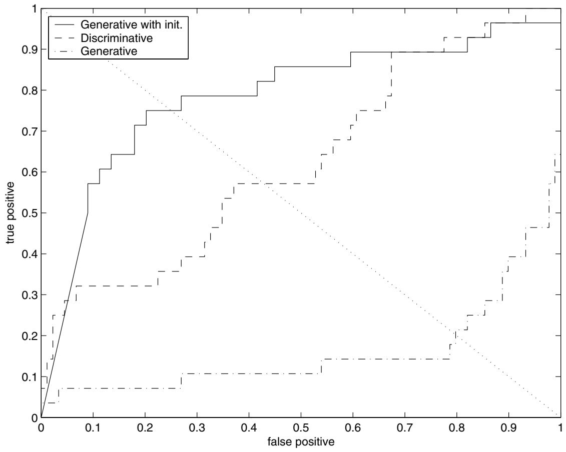

Figure 4. Roc curves of patch labelling.

Table 3. Typical values for speed (sec).

Model	Single train cycle	Total training	Testing
D-L	3	510	0.0015
D-NL	5	625	0.0033
G	386	15440	0.31

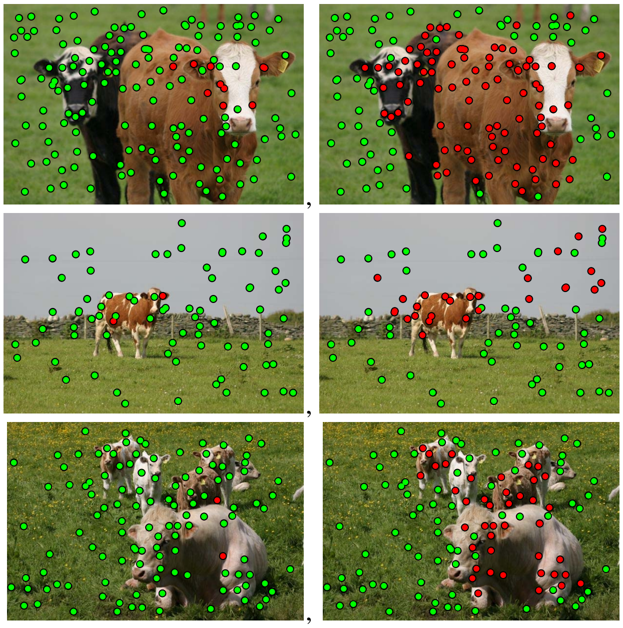

Figure 5. Cow patch labelling examples for discriminative model (left column) and generative model (right column). Red, green and white dots denote cow, background and sheep patches respectively (and are obtained by assigning each patch to the most probable class).

6 Discussion

In this paper we have introduced and compared a generative and a discriminative model for object recognition based on local invariant features. We have shown that the discriminative model is capable of very fast inference, and is able to focus on highly informative features. By contrast the generative model gives high classification accuracy, and also has some ability to localize the objects within the image. However, the generative model is over twenty thousand times slower in classifying new images, and also requires some strongly labelled data for initialization.

One major potential benefit of the generative model is the ability to augment the labelled data with unlabelled data. Indeed, a combination of images which are unlabelled, weakly labelled (having image labels only) and strongly labelled (in which patch labels are also provided as well as the image labels) could be used, provided that all missing variables are 'missing at random'.

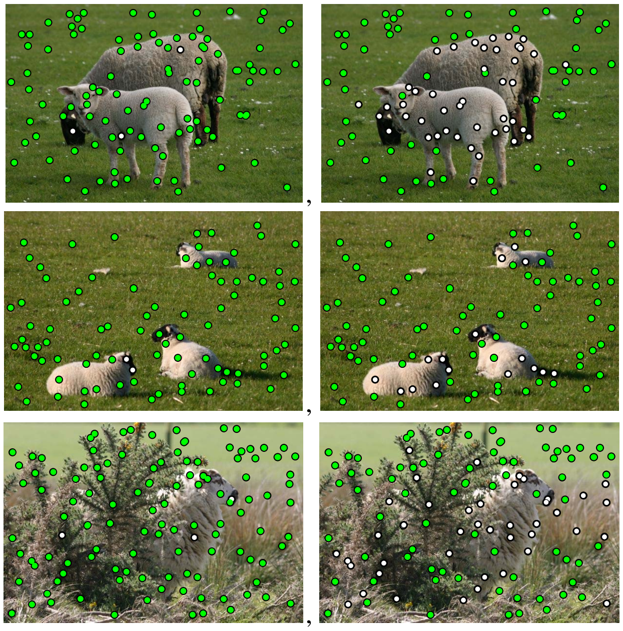

Figure 6. Sheep patch labelling examples for discriminative model (left column) and generative model (right column). Red, green and white dots denote cow, background and sheep patches respectively.

Another significant potential advantage of generative models is the relative ease with which the required invariances can be specified, particularly those arising from geometrical transformations. For instance, the effect of a translation is simply to shift the pixels. By contrast, in a discriminative model ensuring invariance to the resulting highly non-linear transformations of the input variables is non-trivial. However, inference in such a generative model can be very complex due to the need to determine values for the transformation parameters which have high posterior probability, and this generally involves iteration. A discriminative model, on the other hand, is typically extremely fast once trained.

Our investigations suggest that the most fruitful approaches will involve some combination of generative and discriminative models. Indeed, this is already found to be the case in speech recognition where generative hidden Markov models are used to express invariance to non-linear time warping, and are then trained discriminatively by maximizing mutual information in order to achieve high predictive performance.

One promising avenue for investigation is to use a fast discriminative model to locate regions of high probability in the parameter space of a generative model, which can subsequently refine the inferences. Indeed, such coupled generative and discriminative models can mutually train each other, as has already been demonstrated in a simple context in [10].

Finally, for the purposes of this study we have ignored spatial information regarding the relative locations of feature patches in the image. However, most of our conclusions remain valid if a spatial model (such as a Markov random field in the case of a generative model or a conditional random field in the case of a discriminative model) is overlaid on the local classifier.

Acknowledgements

This work was begun while Ilkay Ulusoy was a Visiting Researcher at Microsoft Research Cambridge. The authors would like to thank members of the Machine Learning and Perception Group at Microsoft Research for numerous helpful discussions.

References

[1] K. Barnard, P. Duygulu, D. Forsyth, N. Freitas, D. Blei, and M. I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[2] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[3] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV*, 2004.

[4] G. Dorko and C. Schmid. Selection of scale invariant parts for object class recognition. In *ICCV*, 2003.

[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale invariant learning. In *CVPR*, 2003.

[6] T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[7] D. Lowe. Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[8] O. Maron and A. L. Ratan. Multiple instance learning for natural scene classification. In *CVPR*, 1998.

[9] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86, 2004.

[10] R. Neal P. Dayan, G. E. Hinton and R. S. Zemel. The helmholtz machine. *Neural Computation*, pages 1022–1037, 1995.

[11] M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *ICCV*, 2003.

[12] L. Xie and P. Perez. Slightly supervised learning of part-based appearance models. In *IEEE Workshop on Learning in CVPR*, 2004.