

# Object Recognition via Local Patch Labelling

Christopher M. Bishop<sup>1</sup> and Ilkay Ulusoy<sup>2</sup>

<sup>1</sup> Microsoft Research,  
7 J J Thompson Avenue,  
Cambridge, U.K.

<http://research.microsoft.com/~cmbishop>

<sup>2</sup> METU, Computer Vision and Intelligent Systems Research Lab.  
06531 Ankara, Turkey

<http://www.eee.metu.edu.tr/~ilkay>

**Abstract.** In recent years the problem of object recognition has received considerable attention from both the machine learning and computer vision communities. The key challenge of this problem is to be able to recognize any member of a category of objects in spite of wide variations in visual appearance due to variations in the form and colour of the object, occlusions, geometrical transformations (such as scaling and rotation), changes in illumination, and potentially non-rigid deformations of the object itself. In this paper we focus on the detection of objects within images by combining information from a large number of small regions, or ‘patches’, of the image. Since detailed hand-segmentation and labelling of images is very labour intensive, we make use of ‘weakly labelled’ data in which the training images are labelled only according to the presence or absence of each category of object. A major challenge presented by this problem is that the foreground object is accompanied by widely varying background clutter, and the system must learn to distinguish the foreground from the background without the aid of labelled data. In this paper we first show that patches which are highly relevant for the object discrimination problem can be selected automatically from a large dictionary of candidate patches during learning, and that this leads to improved classification compared to direct use of the full dictionary. We then explore alternative techniques which are able to provide labels for the individual patches, as well as for the image as a whole, so that each patch is identified as belonging to one of the object categories or to the background class. This provides a rough indication of the location of the object or objects within the image. Again these individual patch labels must be learned on the basis only of overall image class labels. We develop two such approaches, one discriminative and one generative, and compare their performance both in terms of patch labelling and image labelling. Our results show that good classification performance can be obtained on challenging data sets using only weak training labels, and they also highlight some of the relative merits of discriminative and generative approaches.

## 1 Introduction

The problem of object recognition 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. Many of

the current approaches to this problem rely on the use of local features obtained from small patches of the image. The motivation for this is that the variability of small patches is much less than that of whole images and so there are much better prospects for generalization, in other words for recognizing that a patch from a test image is similar to patches in the training images. However, the patches must be sufficiently variable, and therefore sufficiently large, to be able to discriminate between the different object categories and also between objects and background clutter. A good way to balance these two conflicting requirements is to determine the object categories present in an image by fusing together partial ambiguous information from multiple patches. Probability theory provides a powerful framework for combining such uncertain information in a principled manner, and will form the basis for our research (the specific local features that we use in this paper are described in Section 2.) Also, the locations of those patches which provide strong evidence for an object also give an indication of the location and spatial extent of that object.

In common with a number of previous approaches, we do not attempt to model the spatial relationship between patches. Although such spatial information is certainly very relevant to the object recognition problem, and its inclusion would be expected to improved recognition performance for many object categories, its role is complementary to that of the texture-like evidence provided by local patches. Here we show that local information alone can already give good discriminatory results.

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. Although the informative features used in [13] are 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 centered on interest points.

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 and the type of detector they used is appropriate for this purpose. Csurka *et al.* [3] used Harris-Laplace interest point operators [9] with SIFT features [7] for the purpose of multi class object category recognition. Features are clustered using K-Means and each feature is labelled according to the closest cluster centre. Histograms of feature labels are then used as class-conditional densities. Since such interest point operators detect many points from the background as well as from the object itself, the features are used collectively to determine the object category, and no information on object localization is obtained. In [4], informative features were selected based on information criteria such as likelihood ratio and mutual information in which DoG and Harris-Laplace interest point detectors with SIFT descriptors were compared. 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/background classification. The two detectors gave similar results although DoG produces more features from the background. Finally, Xie and Perez [14] 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 this paper we develop several new approaches to object recognition based on features extracted from local patches centered on interest points. We begin, in Section 3, by extending the model of [3] which constructs a large dictionary of candidate feature ‘prototypes’. By using the technique of *automatic relevance determination*, our approach can learn which of these prototypes are particularly salient for the problem of discriminating object classes and can thereby give appropriately less emphasis to those which carry little discriminatory information (such as those associated with background clutter). This leads to a significant improvement in classification performance.

While this approach allows the system to focus on the foreground objects, it does not directly lead to a labelling of the individual patches. We therefore develop new probabilistic approaches to object recognition based on local patches in which the system learns not only to classify the overall image, but also to assign labels to patches themselves. In particular, we develop two complementary approaches one of which is discriminative (Section 4) and one of which is generative (Section 5).

To understand the distinction between discriminative and generative, consider a scenario in which an image described by a vector  $\mathbf{X}$  (which might comprise raw pixel intensities, or some set of features extracted from the image) is to be assigned to one of  $K$  classes  $k = 1, \dots, K$ . 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 example 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

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

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

Comparative results from the various approaches are presented in Section 6. These show that the generative approach gives excellent classification performance both for individual patches and for the complete images, but that careful initialization of the

training procedure is required. By contrast the discriminative approach, which gives good results for image labelling but not for patch labelling, is significantly faster in processing test images. Ideas for future work, including techniques for combining the benefits of generative and discriminative approaches, are discussed briefly in Section 7.

## 2 Local Feature Extraction

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 task of detecting and distinguishing cows and sheep in natural images.

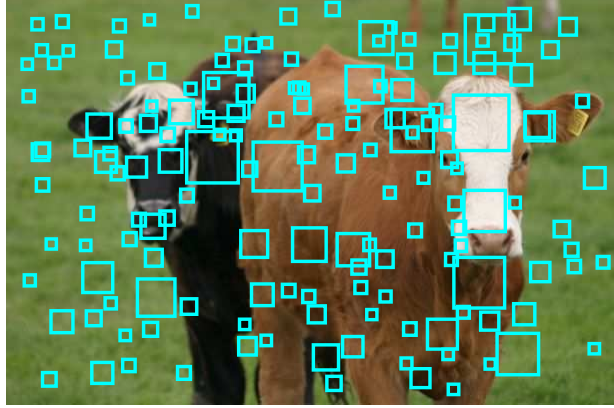
We therefore follow several recent approaches [7, 9] and use an interest point detector to focus attention on a small number of local patches in each image. This is followed by invariant feature extraction from a neighbourhood around each interest point. Specifically we use DoG interest point detectors, and at each interest point we extract a 128 dimensional SIFT feature vector [7] from a patch whose scale is determined by the DoG detector. 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.

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  $k = 1, \dots, K$  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  $j = 1, \dots, J_n$ . 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 (in our example each image contains either cows or sheep).

## 3 Patch Saliency using Automatic Relevance Determination

We begin by considering a simple approach based on [3]. In this method the features extracted from all of the training images are clustered into  $C$  classes using the K-means algorithm, after which each patch in each image is assigned to the closest prototype. Each image  $n$  is therefore described by a fixed-length histogram feature vector  $\mathbf{h}_n$  of length  $C$  in which element  $h_{nc}$  represents the number of patches in image  $n$  which are assigned to cluster  $c$ , where  $c \in \{1, \dots, C\}$  and  $n \in \{1, \dots, N\}$ . These feature



**Fig. 1.** Difference of Gaussian interest points with their local regions, in which the squares are centered at the interest points and the size of the squares indicates the scale of the interest points. The SIFT descriptors and colour features are obtained from these square patches. Note that interest points fall both on the objects of interest (the cows) and also on the background.

vectors are then used to construct a classifier which takes an image  $\mathbf{X}_n$  as input, converts it to a feature vector  $\mathbf{h}_n$  and then assigns this vector to an object category. Here the assumption is that each image belongs to one and only one of some number  $K$  of mutually exclusive classes. In [3] the classifier was based either on naive Bayes or on support vector machines.

Here we use a linear softmax model since this can be readily extended to determine feature saliency as discussed shortly. Thus the model computes a set of outputs given by

$$y_k(\mathbf{h}_n, \mathbf{w}) = \frac{\exp(\mathbf{w}_k^T \mathbf{h}_n)}{\sum_l \exp(\mathbf{w}_l^T \mathbf{h}_n)} \quad (2)$$

where  $k \in \{1, \dots, K\}$ . Here the quantity  $y_k(\mathbf{h}_n, \mathbf{w})$  which can be interpreted as the posterior probability that image vector  $\mathbf{h}_n$  belongs to class  $k$ . The parameter vector  $\mathbf{w} = \{\mathbf{w}_k\}$  is found by maximum likelihood using iterative re-weighted least squares [10]. We shall refer to this approach as VQ-S for vector quantized softmax. Results from this method will be presented in Section 6.

An obvious problem with this approach is that the patches which contribute to the feature vector come from both the foreground object(s) and also from the background. Changes to the background cause changes in the feature vector even if the foreground object is the same. Furthermore, some foreground patches might occur on objects from different classes, and are therefore provide relatively little discriminatory information compared to other patches which are more closely associated with particular object categories.

We can address this problem using the Bayesian technique of *automatic relevance determination* or *ARD* [8]. This involves the introduction of a prior distribution over the parameter vector  $\mathbf{w}$  in which each input variable  $h_c$  has a separate hyperparameter  $\alpha_c$  corresponding to the inverse variance (or precision) of the prior distribution of the weights  $\mathbf{w}_c$  associated with that input, so that

$$p(\mathbf{w}|\alpha) = \prod_{c=1}^C \mathcal{N}(\mathbf{w}_c|\mathbf{0}, \alpha_c^{-1}\mathbf{I}). \quad (3)$$

During learning the hyperparameters are updated by maximizing the marginal likelihood, i.e. the probability of the training labels  $D$  given  $\alpha$  in which  $\mathbf{w}$  has been integrated out, given by

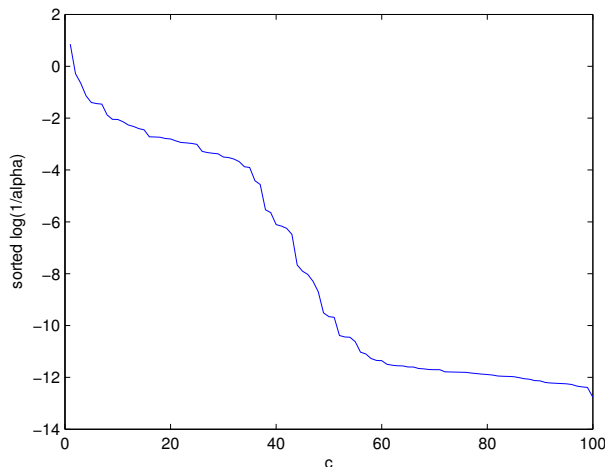
$$p(D|\alpha) = \int p(D|\mathbf{w})p(\mathbf{w}) d\mathbf{w}. \quad (4)$$

This is known as the *evidence procedure* and the values of the hyperparameters found at convergence express the relative importance of the input variables in determining the image class label. Specifically, the hyperparameters represent the inverse variances of the weights, and so a large value of  $\alpha_c$  implies that the corresponding parameter vector  $\mathbf{w}_c$  has a distribution which is concentrated around zero and so the associated input variable  $h_c$  has little effect in determining the output values  $y_k$ . Such inputs have low relevance. By contrast a high value of  $\alpha_c$  corresponds to an input  $h_c$  whose value plays an important role in determining the class label. The inclusion of ARD leads to an improvement in classification performance, as discussed in Section 6. We shall refer to this model as VQ-ARD.

With this approach we can rank the patch clusters according to their relevance. The logarithm of the inverse of the hyperparameter  $\alpha_c$  is sorted and plotted in Figure 2. Equivalently this can be plotted as a histogram of  $\alpha_c$  values, as shown in Figure 3. It is interesting to note that in this problem the hyperparameter values form two groups in which one group can loosely be considered as relevant and the other as not relevant, so far as the discrimination task is concerned.

Figure 4 shows the properties of the most relevant cluster and of the least relevant cluster, as well as that of an intermediate cluster, according to the ARD analysis based on  $C = 100$  cluster centers. Note that the images have been hand segmented in order to identify the foreground region. This segmentation is used purely for test purposes and plays no role during training. The top row shows the features belonging to the worst cluster, i.e. ranked 100, on a sheep image and on a cow image. This feature exists in both classes and thus provides a little information to make a classification. The middle row shows the locations of patches assigned to the cluster which is ranked 27, in which we see that all of the patches belong to the background. Finally, the bottom row of the figure shows the features belonging to the most relevant cluster, ranked 1, on the same sheep and cow images. This feature is not observed on the sheep image but there are several patches assigned to this cluster on the cow image. Thus the detection of this feature is a good indicator of the presence of a cow.

It is also interesting to explore the behaviour of the two groups of clusters corresponding to the two modes in the distribution of hyper-parameter values shown in



**Fig. 2.** The sorted values of the log variance (inverse of the hyperparameter  $\alpha$ ).

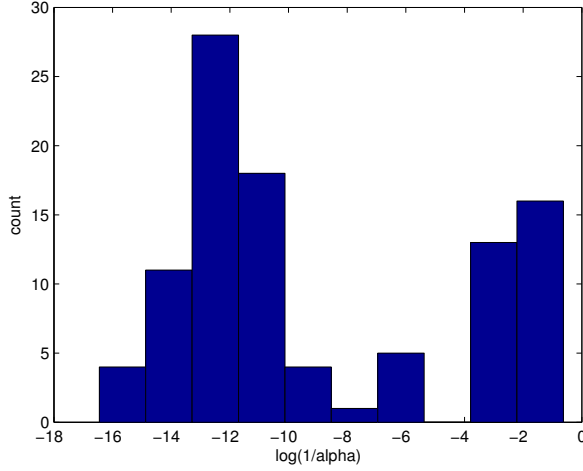
Figure 3. Figure 5 shows examples of cow and sheep images in each case showing the locations of the clusters associated with the two modes.

Although this approach is able to focuss attention on foreground regions, we have seen that not all foreground patches have high saliency, and so this approach cannot reliably identify regions occupied by the foreground objects. We therefore turn to the development of new models in which we explicitly consider the identity of individual patches and not simply their saliency for overall image classification. In particular the hard quantization of K-means is abandoned in favour of more probabilistic approaches. First we discuss a discriminative model and then we turn to a complementary generative model.

#### 4 The Discriminative Model with Patch Labelling

Since our goal is to determine the class membership of individual patches, we associate with each patch  $j$  in an 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. Note that this assumption is not essential, and other formulations could also be considered. These components can be grouped together into vectors  $\tau_{nj}$ . 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  $\{\tau_{nj}\}$  labels are hidden (latent) variables, which of course makes the training problem much harder.

We now introduce a discriminative model, which corresponds to the directed graph shown in Figure 6.



**Fig. 3.** The histogram of the log variances.

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

$$y_k(\mathbf{x}_j, \mathbf{w}) = \frac{\exp(\mathbf{w}_k^T \mathbf{x}_j)}{\sum_l \exp(\mathbf{w}_l^T \mathbf{x}_j)} \quad (5)$$

which satisfy  $0 \leq y_k \leq 1$  and  $\sum_k y_k = 1$ . More generally we can use a multi-layer neural network, a relevance vector machine, or any other parametric model that gives probabilistic outputs and which can be optimized using gradient-based methods. The probability of a patch label  $\tau_j$  is then given by

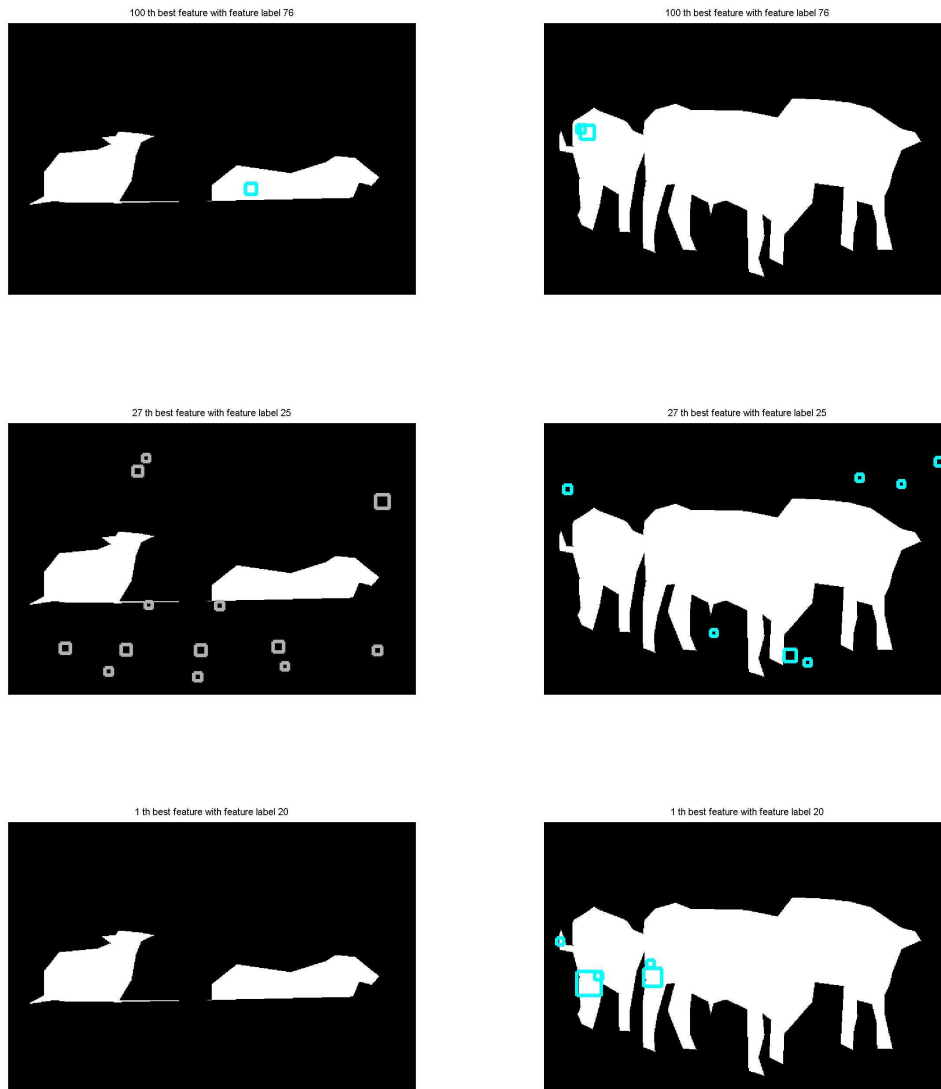
$$p(\tau_j | \mathbf{x}_j) = \prod_{k=1}^K y_k(\mathbf{x}_j, \mathbf{w})^{\tau_{jk}} \quad (6)$$

where the binary exponent  $\tau_{jk}$  simply pulls out the required term (since  $y_k^0 = 1$  and  $y_k^1 = y_k$ ).

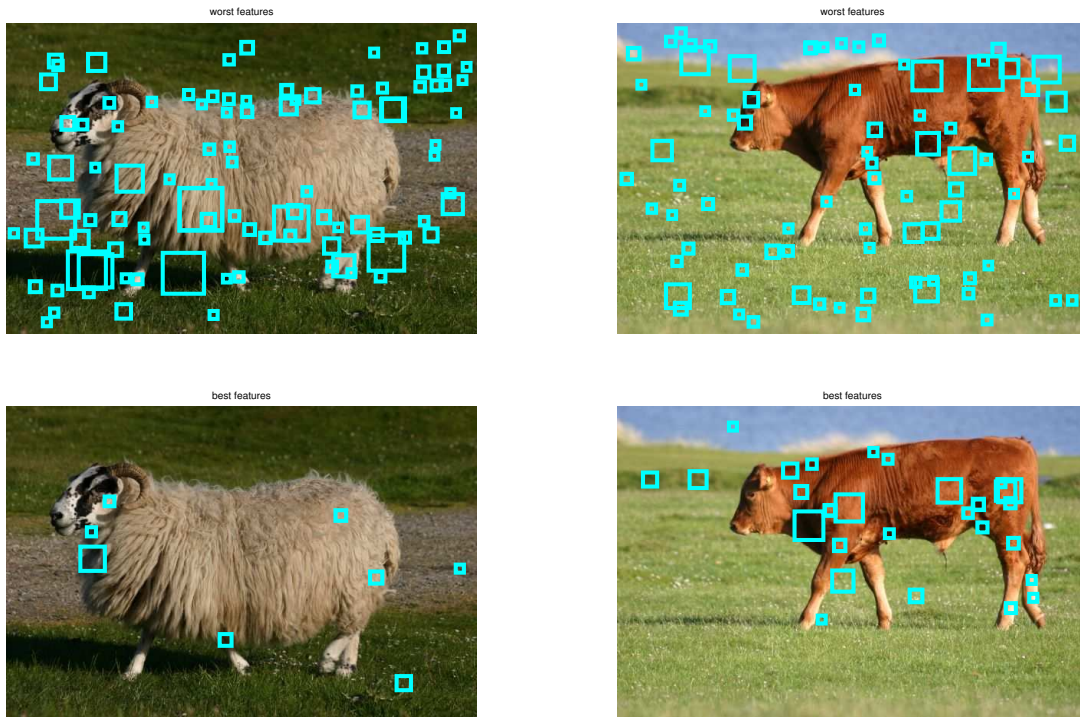
Next we assume that if one, or more, of the patches carries the label for a particular class, then the whole image will. For instance, if there is at least one local patch in the image which is labelled ‘cow’ then the whole image will carry a ‘cow’ label (recall that an image can carry more than one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

$$p(\mathbf{t} | \boldsymbol{\tau}) = \prod_{k=1}^K \left[ 1 - \prod_{j=1}^J [1 - \tau_{jk}] \right]^{t_k} \left[ \prod_{j=1}^J [1 - \tau_{jk}] \right]^{1-t_k}. \quad (7)$$

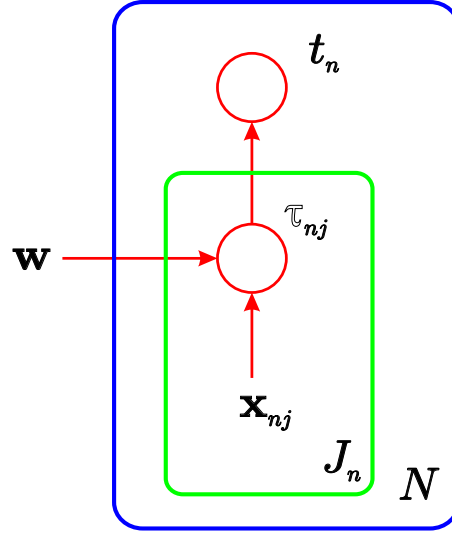




**Fig. 4.** The top row shows example cow and sheep images, with the foreground regions segmented, together with the locations of patches assigned to the least relevant (ranked 100) cluster center. Similarly the middle row analogous results for a cluster of intermediate relevance (ranked 27) and the bottom row shows the cluster assignments for the most relevant cluster (ranked 1). The centers of the squares are the locations of the patches from which the features are obtained and the size of the squares show the scale of the patches.



**Fig. 5.** Illustration of the behaviour of the two modes in the histogram of hyper-parameter values seen in Figure 5. The left column shows a typical example from the sheep class while the right column shows a typical example from the cow class. In the top row the squares denote the locations of interest points assigned to clusters in the left hand mode of the histogram corresponding to low relevance clusters, while the bottom row gives the analogous results to the high relevance model. The threshold between high and low was set by eye to  $\ln(1/\alpha) = -5$ . Note that the high relevance clusters are associated predominantly with the foreground, while the low relevance ones occur on both the foreground and the background.



**Fig. 6.** Graphical representation of the discriminative model for object recognition.

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 due to the factorization implied by the graph in Figure 6 to give

$$\begin{aligned}
 p(\mathbf{t}|\mathbf{X}) &= \sum_{\boldsymbol{\tau}} \left\{ p(\mathbf{t}|\boldsymbol{\tau}) \prod_{j=1}^J p(\tau_j|\mathbf{x}_j) \right\} \\
 &= \prod_{k=1}^K \left[ 1 - \prod_{j=1}^J [1 - y_k(\mathbf{x}_j, \mathbf{w})] \right]^{t_k} \left[ \prod_{j=1}^J [1 - y_k(\mathbf{x}_j, \mathbf{w})] \right]^{1-t_k}. \quad (8)
 \end{aligned}$$

This can be viewed as a softened (probabilistic) version of the logical 'OR' function [12].

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

$$E(\mathbf{w}) = - \sum_{n=1}^N \sum_{k=1}^C \{ t_{nk} \ln [1 - Z_{nk}] + (1 - t_{nk}) \ln Z_{nk} \} \quad (9)$$

where we have defined

$$Z_{nk} = \prod_{j=1}^{J_n} [1 - y_k(\mathbf{x}_{nj}, \mathbf{w})]. \quad (10)$$

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  $\mathbf{w}_{\text{ML}}$  is found, the corresponding functions  $y_k(\mathbf{x}, \mathbf{w}_{\text{ML}})$  for  $k = 1, \dots, K$  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 6.

## 5 The Generative Model with Patch Labelling

Next we turn to a description of our generative model, whose graphical representation is shown in Figure 7. The structure of this model mirrors closely that of the discriminative

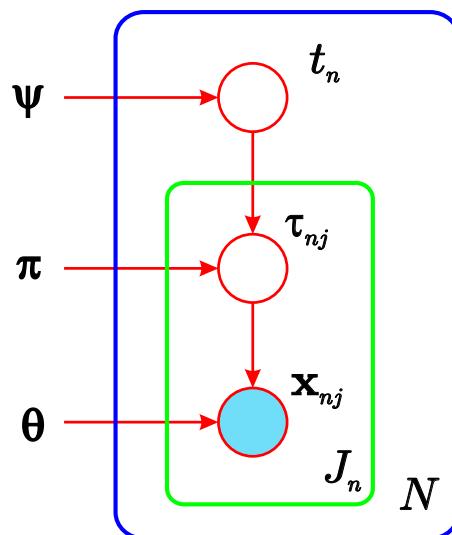


Fig. 7. Graphical representation of the generative model for object recognition.

model. In particular, the same class-label variables  $\tau_{nj}$  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 to determine  $p(\mathbf{X}|\mathbf{t})$  without concluding that the prior probabilities  $p(\mathbf{t})$  of objects is small.

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

$$p(\mathbf{t}) = \prod_{k=1}^K \psi_k^{t_k} (1 - \psi_k)^{1-t_k}. \quad (11)$$

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  $\pi_k$ , one for each class, such that  $\pi_k \geq 0$ , constrained by the subset of classes actually present in the image. Thus

$$p(\boldsymbol{\tau}_j|\mathbf{t}) = \left( \sum_{l=1}^K t_l \pi_l \right)^{-1} \prod_{k=1}^K (t_k \pi_k)^{\tau_{jk}}. \quad (12)$$

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  $k$ , the distribution of the patch feature vector  $\mathbf{x}$  is governed by a separate mixture of Gaussians which we denote by  $\phi_k(\mathbf{x}; \boldsymbol{\theta}_k)$ , so that

$$p(\mathbf{x}_j|\boldsymbol{\tau}_j) = \prod_{k=1}^K \phi_k(\mathbf{x}_j; \boldsymbol{\theta}_k)^{\tau_{jk}} \quad (13)$$

where  $\boldsymbol{\theta}_k$  denotes the set of parameters (means, covariances and mixing coefficients) associated with this mixture model, and again the binary exponent  $\tau_{jk}$  simply picks out the required class.

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

$$\prod_{n=1}^N p(\mathbf{t}_n) \prod_{j=1}^{J_n} [p(\mathbf{x}_{nj}|\boldsymbol{\tau}_{nj})p(\boldsymbol{\tau}_{nj}|\mathbf{t}_n)]. \quad (14)$$

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

$$\sum_{n=1}^N \sum_{j=1}^{J_n} \left\{ \sum_{k=1}^K \langle \tau_{nj k} \rangle \ln [t_{nk} \pi_k \phi_k(\mathbf{x}_{nj})] - \ln \left( \sum_{l=1}^K t_{nl} \pi_l \right) \right\}. \quad (15)$$

In the E-step the expected values of  $\tau_{nkj}$  are computed using

$$\langle \tau_{nkj} \rangle = \sum_{\{\boldsymbol{\tau}_{nj}\}} \tau_{nkj} p(\boldsymbol{\tau}_{nj} | \mathbf{x}_{nj}, \mathbf{t}_n) = \frac{t_{nk} \pi_k \phi_k(\mathbf{x}_{nj})}{\sum_{l=1}^K t_{nl} \pi_l \phi_l(\mathbf{x}_{nj})}. \quad (16)$$

Notice that the first factor on the right hand side of (12) has cancelled in the evaluation of  $\langle \tau_{nkj} \rangle$ .

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

$$\pi_k = \left[ \sum_{n=1}^N J_n t_{nk} \left( \sum_{l=1}^K t_{nl} \pi_l \right)^{-1} \right]^{-1} \sum_{n=1}^N \sum_{j=1}^{J_n} \langle \tau_{nkj} \rangle. \quad (17)$$

Since these represent coupled equations we perform several (fast) iterations of these equations before proceeding with the next EM cycle (note that for this purpose the sums over  $j$  can be pre-computed since they do not depend on the  $\{\pi_k\}$ ).

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

$$\sum_{n=1}^N \sum_{j=1}^{J_n} \langle \tau_{nkj} \rangle \ln \phi_k(\mathbf{x}_{nj}; \boldsymbol{\theta}_k) + \text{const}. \quad (18)$$

This is easily maximized for each class  $k$  separately using the EM algorithm (in an inner loop), since (18) simply represents a log likelihood function for a weighted data set in which patch  $(n, j)$  is weighted with  $\langle \tau_{nkj} \rangle$ . Specifically, we use a model in which  $\phi_k(\mathbf{x}; \boldsymbol{\theta}_k)$  is given by a Gaussian mixture distribution of the form

$$\phi_k(\mathbf{x}; \boldsymbol{\theta}_k) = \sum_{m=1}^M \rho_{km} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}). \quad (19)$$

The E-step is given by

$$\gamma_{njkm} = \frac{\rho_{km} \mathcal{N}(\mathbf{x}_{nj} | \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km})}{\sum_{m'} \rho_{km'} \mathcal{N}(\mathbf{x}_{nj} | \boldsymbol{\mu}_{km'}, \boldsymbol{\Sigma}_{km'})} \quad (20)$$

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

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

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. Such 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 [14] 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 [13] 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  $\tau_j$  for patch  $j$  can be found by marginalizing out all other hidden variables

$$\begin{aligned} p(\tau_j | \mathbf{X}) &= \sum_{\mathbf{t}} \sum_{\boldsymbol{\tau} / \tau_j} p(\boldsymbol{\tau}, \mathbf{X}, \mathbf{t}) \\ &= \sum_{\mathbf{t}} p(\mathbf{t}) \frac{1}{\left( \sum_{l=1}^K \pi_l t_l \right)^J} \prod_{k=1}^K (\pi_k t_k \phi_k(\mathbf{x}_j))^{\tau_j k} \prod_{i \neq j} \left[ \sum_{k=1}^K \pi_k t_k \phi_k(\mathbf{x}_i) \right] \end{aligned} \quad (21)$$

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

For the inference of image label we require the posterior probability of image label  $\mathbf{t}$ , which can be computed using

$$p(\mathbf{t} | \mathbf{X}) \propto p(\mathbf{X} | \mathbf{t}) p(\mathbf{t}) \quad (22)$$

in  $p(\mathbf{t})$  is computed from the coefficients  $\{\psi_k\}$  for each setting of  $\mathbf{t}$  in turn, and  $p(\mathbf{X} | \mathbf{t})$  is found by summing out patch labels

$$p(\mathbf{X} | \mathbf{t}) = \sum_{\boldsymbol{\tau}} \prod_{j=1}^J p(\mathbf{X}, \tau_j | \mathbf{t}) = \prod_{j=1}^{J_n} \frac{\sum_{k=1}^K t_k \pi_k \phi_k(\mathbf{x}_j)}{\sum_{l=1}^K t_l \pi_l}. \quad (23)$$

## 6 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 (5) with 144 inputs, corresponding to the 144 features discussed in Section 2 and 3 outputs (cow, sheep, background). We also explore 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 [14]). To test this we used some segmented images for initialization purposes (but not for optimization). 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 the 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. Similarly, covariance matrices were computed using the data points assigned to the respective center.

In the test phase of both discriminative and generative models, we input the patch features to the models and obtain the posterior probabilities of the patch labels as the outputs using (5) for discriminative model and (21) for the generative model. The posterior probability of the image label is computed as in (8) for the discriminative model and (22) for the generative case. We can therefore investigate the ability of the two models both to predict the class labels of whole images and of their constituent patches. The latter is important for object localization.

The overall correct rates of object recognition, i.e. image labelling, is given in Table 1 for the VQ-S, VQ-ARD, linear discriminative (D-L), nonlinear discriminative (D-NL) and generative (G) models.

**Table 1.** Overall correct rates.

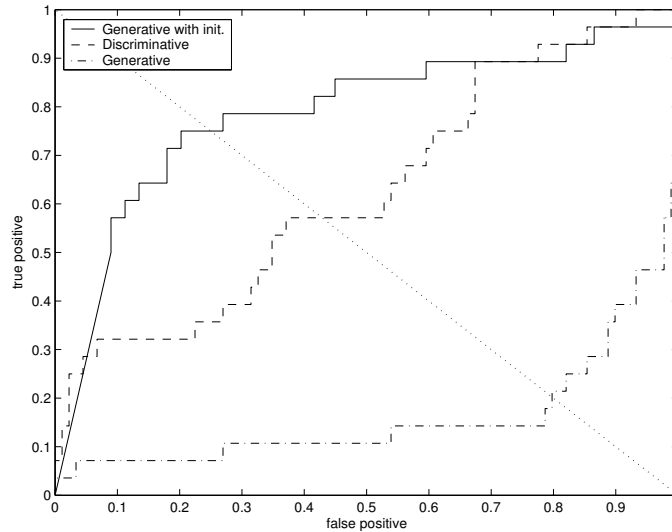
VQ-S	VQ-ARD	D-L	D-NL	G
80%	92%	82.5%	87.2%	97%

It is also interesting to investigate the extent to which the discriminative and generative 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 8. We also plot the ROC curve for the generative model when random initialization is performed to show the importance of initialization for such models.



**Table 2.** Patch labelling scores.

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



**Fig. 8.** ROC curves of patch labelling.

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 the accurate labelling of both foreground and background patches. Some examples of patch labelling for test images are given in Figure 9 for cow images and in Figure 10 for sheep images.

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 200 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 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

## 7 Discussion

In this paper we have introduced and compared a variety of local patch-based models for object recognition. We have shown that automatic relevance determination allows a system to learn which features are most salient in determining the present of an object. We have also introduced novel discriminative and generative models which have complementary strengths and limitations, and shown that the discriminative model is capable of fast inference, and is able to focus on highly informative features, while the generative model gives high classification accuracy, and also has some ability to localize the objects within the image. However, the generative model requires careful initialization in order to achieve good results.

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’.

Another significant potential advantage of generative models is the relative ease with which 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 very 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 [11].

One of the limitations of the techniques discussed here is the use of interest point detectors that are not tuned to the problem being solved (since they are hand-crafted rather than learned) and which are therefore unlikely in general to focus on the most discriminative regions of the image. Similarly, the invariant features used in our study

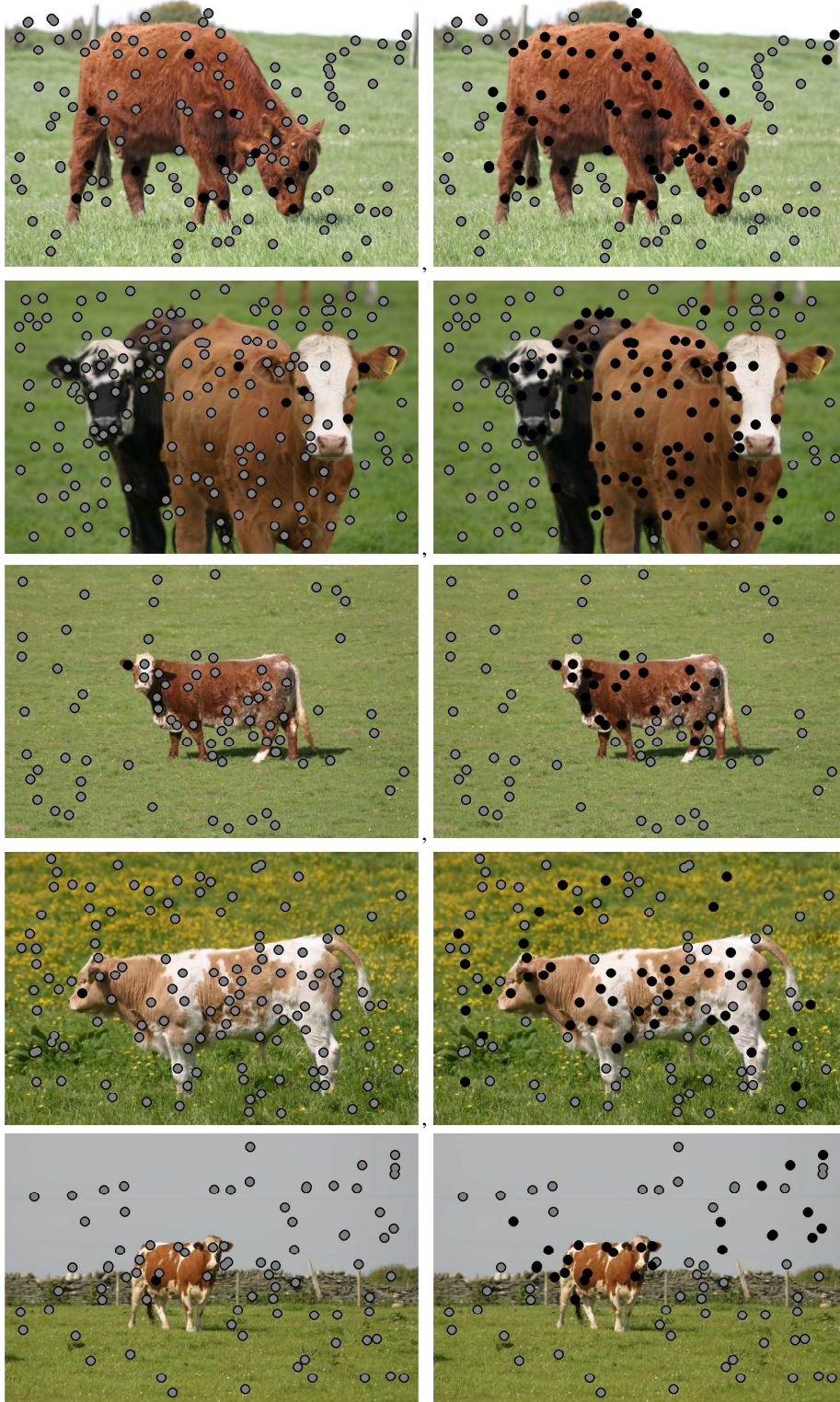
were hand-selected. We expect that robust recognition of a large class of object categories will require that local features be learned from data.

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 is combined with the local information provided by the patch features.

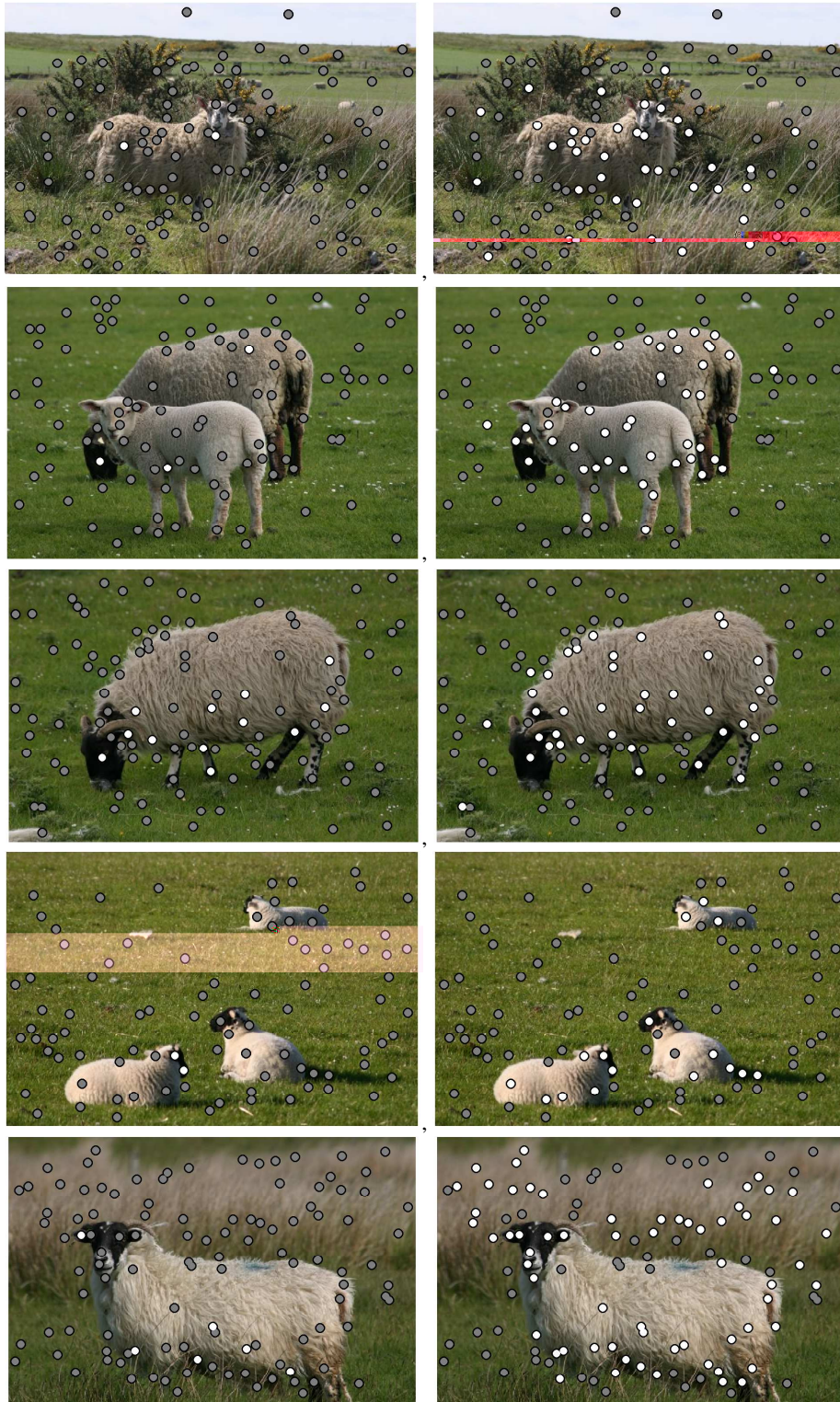
**Acknowledgements** We would like to thank Antonio Criminisi, Geoffrey Hinton, Fei Fei Li, Tom Minka, Markus Svensen and John Winn for numerous discussions.

## 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 *Workshop on Statistical Learning in Computer Vision, 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. D. J. C. MacKay. Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks. 6(3):469–505, 1995.
9. K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86, 2004.
10. I. T. Nabney. *Netlab Algorithms for Pattern Recognition*. Springer, 2004.
11. R. Neal P. Dayan, G. E. Hinton and R. S. Zemel. The helmholtz machine. *Neural Computation*, pages 1022–1037, 1995.
12. J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, 1998.
13. M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *ICCV*, 2003.
14. L. Xie and P. Perez. Slightly supervised learning of part-based appearance models. In *IEEE Workshop on Learning in CVPR*, 2004.



**Fig. 9.** Cow patch labelling examples for discriminative model (left column) and generative model (right column). Black, gray and white dots denote cow, background and sheep patches respectively (and are obtained by assigning each patch to the most probable class).



**Fig. 10.** Sheep patch labelling examples for discriminative model (left column) and generative model (right column). Black, gray and white dots denote cow, background and sheep patches respectively.